Self Study

Computer Science

Computer Science
The Erik Jonsson School of Engineering and Computer Science
The University of Texas at Dallas
Richardson, Texas 75080
http://www.utdallas.edu/dept/cs/

2005

Report prepared for the ABET
Computing Accreditation Commission

SELF-STUDY QUESTIONNAIRE
FOR THE REVIEW
OF COMPUTER SCIENCE PROGRAM

Submitted by
The University of Texas at Dallas
To the Computing Accreditation Commission

The Erik Jonsson School of
Engineering and Computer Science
The University of Texas at Dallas
Richardson, Texas 75080

PRIMARY CONTACT: Dr. Gopal Gupta
Phone: 972-883-4107; Fax: 972-883-2399; Email: gupta@utdallas.edu

Computing Accreditation Commission
Accreditation Board for Engineering and Technology, Inc.
111 Market Place, Suite 1050
Baltimore, Maryland 21202-4012
Phone: 410-347-7700
Fax: 410-625-2238
e-mail: cac@abet.org
www: http://www.abet.org/

6/22/2005

The Future of the University of Texas at Dallas

Dean's Message

The Erik Jonsson School of Engineering and Computer Science is located in the heart of the **"Telecom Corridor®**". The Jonsson School focused, from its inception, on educating electrical engineers and computer scientists to serve the needs of local high-tech industry. In recent years, rapid growth in enrollments plus increased specialization have sparked the development of new degree programs in telecommunication engineering, the nation's first such accredited program, computer engineering, and software engineering. Together with the traditional EE and CS disciplines, these programs currently form the core of the School's academic curricula.

The Jonsson School and UTD stand poised for a new era of growth and excellence, based in large part on a 5-year, $300 million initiative involving the State of Texas, the UT System, and Texas Instruments. The plan, announced in June 2003, will significantly enhance both the breadth and quality of engineering education and research at UTD. Included is the construction of a 200,000 sq. ft. state-of-the-art research center, to be completed in 2006. This new facility will broaden the scope of current research activities and, more importantly, foster new interdisciplinary programs with related sciences such as physics, chemistry, and molecular biology, the UTD School of Management, and even the arts and humanities. It will also facilitate joint research with other outstanding local institutions such as UT Southwestern Medical Center.

The need for interdisciplinary programs has been driven by the rapid convergence of fields such as microelectronics, sensors, nanoscience, biotechnology, information science, and environmental and health sciences. As the boundaries between these disciplines become increasingly blurred, emerging technologies will rely more heavily on our ability to integrate them into a coherent engineering system. Most recently, we began a new initiative called "To 50 in five. With our community and corporate partnerships, UTD is poised to move into its next level of development. The Jonsson School is working tirelessly in conjunction with university officials to crack the ranks of the top 50 engineering programs and the top academic research institutions nationally within the next five years. We are confident our effort will be successful.

These are exciting times for all of us at the Jonsson School. The opportunity to have an impact on the future of technology has never been greater. We are inviting all that have the interest and the desire to join us at UTD and play a part in forging that future.

C. Robert Helms
The Erik Jonsson School of Engineering and Computer Science
The University of Texas at Dallas
2601 North Floyd Road
Richardson, Texas 75083

Introduction

The University of Texas at Dallas was founded in 1969 and has quickly evolved into the leading institution in the Dallas area. The University has experienced tremendous growth over the past several years. Student enrollment reached 13,316 and faculty size reached 697 (472 full-time) in Fall 2004. At the same time, UTD has remained highly selective with its freshman class consistently ranking among the top three in the State of Texas in terms of SAT scores (average of 1239 for the Freshman class in Fall 2004), number of National Merit Scholars (38 in Fall 2004), and students in the top 10% of their class. A building boom since 2001 added over 500,000 square feet of new academic space. A new $85 million (200,000 sq. ft.) Natural Science and Engineering research facility is expected by Fall 2006.

The Erik Jonsson School of Engineering and Computer Science was founded in 1986 and moved into a new facility in 1992. The School has a leading role in achieving the stated mission of UTD "to be a nationally recognized top-tier University sculpted within a model of focused excellence." The Fall 2002 enrollment of 3,624 for the School represents a 371% increase since 1992 (976). Enrollment dropped to 3,394 in Fall 2003 and 2,854 in Fall 2004 but it is expected to stabilize in Fall 2005 and increase at modest rates in the next few years. The original programs in Computer Science and Electrical Engineering have expanded to include degrees in Telecommunications Engineering (the first in the nation), Computer Engineering, and Software Engineering. The Software Engineering program has risen rapidly, reflecting a local demand for people educated in the field despite the recent economic downturn. Additional programs in Material Sciences and Bioengineering are expected soon.

The Department of Computer Science has experienced explosive growth in the last 15 years. Student population has increased from a few hundred in 1992 to a maximum of 1375 in 2001. Since then it has declined to 1002 in Fall 2004, however, it is expected to stabilize in Fall 2005 as well. Faculty size has more than quadrupled in the last 15 years: from 13 faculty members in 1991 to 56 today. This growth in faculty size continues. In Fall 2002, the Department of Computer Science moved to a new 152,000 sq. ft. building with 10 modern classrooms. The addition of the new building more than doubled the space available to the Computer Science department. The number of degree offerings has grown as well. Today the department offers a BS, MS, and Ph.D in Computer Science and Software Engineering, as well as interdisciplinary degrees in Telecom Engineering and Computer Engineering (jointly with the Electrical Engineering Department).

While the department has been making rapid strides in imparting education in computer science and software engineering, it has been developing cutting edge research programs at a similar pace. A number of internationally recognized research groups and centers have been set up at UTD in recent years. These efforts culminated with the Jonsson School Research Excellence (JSRE) Initiative through which the state of Texas will invest $300 Million in Engineering, Science, and Computer Science research and education at UTD. The JSRE Initiative project, that began in 2003, aims to make UT Dallas an international powerhouse in both education and research.

Table of Contents

I. Objectives and Assessments

INTENT: The program has documented, measurable objectives, including expected outcomes for graduates. The program regularly assesses its progress against its objectives and uses the results of the assessments to identify program improvements and to modify the program's objectives.

The ***Intent*** must be met in order for a program to be deemed accreditable. One way to meet the ***Intent*** of this criterion is to satisfy each one of the **Standards** listed below. To do this, answer the questions associated with the **Standards**. If one or more **Standards** are not satisfied, it is incumbent upon the institution to demonstrate and document clearly and unequivocally how the ***Intent*** is met in some alternative fashion.

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

Standard I-1. The program must have documented, measurable objectives.

Computer Science at UT-Dallas started as a Major within the Dept. of Mathematical Sciences in the School of Natural Sciences in 1975. As was the case for UT Dallas as a whole, the emphasis in the early years was on graduate education and research. The first two doctoral degrees in Computer Science were granted in 1980 with the recipients going to successful academic careers at UT Austin and the University of Toronto. Computer Science steadily grew in size and independence culminating with its move to the new School of Engineering and Computer Science in 1986.

While retaining quality graduate education and research among its goals, the importance of undergraduate education at UT-Dallas grew steadily as well, from upper-level only programs in the late 70s to full 4-yr undergraduate programs in 1990. With admission requirements set at the levels of UT-Austin, UT-Dallas' high quality student body experienced tremendous growth (from 4,845 in 1990 to 13,316 in 2004) with the changes particularly striking within the School of Engineering and Computer Science which saw undergraduate enrollments rise from 470 in 1990 to 1,840 in 2004. For several years in a row now, the Department of Computer Science has ranked first nationally in the number of degrees (BS, MS and PhD) granted.

At present, providing high quality educational programs that enable undergraduates to succeed in industry, government, and/or graduate education is the Computer Science Department's principal objective and a critical component of the School's mission and planning. This was re-affirmed over the past year by Dean Helms, the faculty, and the School's Advisory Board. The mission, goals, objectives and outcomes of the Computer Science program are described in the next section within the context of the mission and goals of the University and the School of Engineering and Computer Sciences. They are documented and publicized in the University catalog, web pages, brochures, etc.

Standard I-2. The program's objectives must include expected outcomes for graduating students.

A. Objectives.

Please attach items that support or precede the objectives, e.g.,

- mission statements from institution, college, department, program
- plans (institution, college, department, etc.)
- all objectives including student outcomes (itemize)
- process for assessments
- who is involved in assessment and improvement?
- data from assessments
- inputs from any supporting Office of Assessment

1. Indicate below or attach to this document your educational objectives for this program. These objectives must include expected outcomes for graduating students.

UTD Mission Statement

The mission of The University of Texas at Dallas is to provide Texas and the nation with the benefits of educational and research programs of the highest quality. These programs address the multi-dimensional needs of a dynamic, modern society driven by the development, diffusion, understanding and management of advanced technology.

Strategic Intent:

To be a nationally recognized top-tier university sculpted within a model of focused excellence. The university emphasizes education and research in engineering, science, technology and management while maintaining programs of focused excellence in other academic areas. Within the context of this mission, the goals of the university are as follows:

- To provide able, ambitious students with a high-quality, cost-effective education that combines the nurturing environment of a liberal arts college with the intellectual rigor and depth of a major research university.

- To discover new knowledge and to create new art that enriches civilization at large and contributes significantly to economic and social programs.

- To enhance the productivity of business and government with strategically designed, responsively executed programs of research, service and education.

The university intends to achieve these objectives by investing in students and faculty, building upon its programs, policies and operations and enhancing institutional character and excellence in education. The majors points of UTD's strategic plan to accomplish these goals are as follows:

- Continue to strengthen the identity of the university as a leader in higher education in terms of excellent faculty and superior students.

- Enhance the quality of its students' learning experiences and its employees' work environment.

- Emphasize education and research in science and technology and in leadership and management, while maintaining concurrent programs of focused excellence in other fundamental fields of art and knowledge.

- Expand and intensify partnerships relations with business, governmental and educational neighbors.

- Enhance programmatic quality and institutional balance while adhering to rigorous quality standards.

- Actively pursue external support of and funding for the ambitious academic and service programs integral to its mission.

Mission Statement of the Erik Jonsson School of Engineering and Computer Science

A new mission statement for the Erik Jonsson School (EJS) of Engineering and Computer Science was prepared in the Spring of 2004. This new mission endeavors to:

- Deliver a state of the art high technology engineering education for Dallas & Collin Counties, the DFW Metroplex, and the State of Texas. This goal is to be achieved by developing highly effective B.S. & M.S. Coursework Degree programs as well as M.S. and Ph.D. Thesis Degree programs. The EJS school aspires to impart knowledge in a way that will produce “agile” students with innovative and entrepreneurial skills.
- Create new state of the art engineering knowledge through research & technology transfer. The research produced will be the outcome of M.S. and Ph.D. Theses.
- Develop partnerships with government and the private sector to apply new knowledge for economic growth and high tech job creation in order to strengthen existing regional firms, promote the growth of new regional firms, as well as create new high paying private sector jobs.
- Provide leadership and outreach to nurture tomorrow’s leaders in science, mathematics, and high technology education and business

A concrete goal of the the Erik Jonsson School is to be rated one of the top 50 engineering schools in the country within 5 years. Considerable resources and efforts are being invested to reach this goal. This includes the JSRE Initiative through which the state of Texas will invest $300 million in education and research in engineering, science and computer science during the period 2003-2008.

The previous mission statement of the Erik Jonsson School (EJS) was:

> To play a distinctive and productive role in engineering and computer science, and deliver value to our students and research sponsors, by closing the gap between academic research and industrial practice.
>
> To achieve excellence by recruiting faculty members who are outstanding in research and who are able and willing to collaborate with others in academia, industry and government, and by maintaining high standards for students and for faculty promotion and tenure.
>
> Reputation will follow from real accomplishments that result from following a distinctive path based on our unique situation, history and opportunities.

While developing its curriculum, the Jonsson School is particularly aware of the perceived future needs of the industries of North Texas related to information systems and electronics manufacturing. The engineering programs prepare individuals for direct entry at the baccalaureate level into professional practice, but the program emphasizes a strong analytical preparation for continued formal education at the masters and doctoral level. A specific mission of UT Dallas and the Jonsson School is to provide opportunities for persons employed full time in local industry to continue and complete their education at both undergraduate and graduate levels. The Jonsson School also strives to use modern computing and telecommunications technology to enhance the quality of education.

Mission Statement of the Department of Computer Science

The mission of the Department of Computer Science is to prepare undergraduate and graduate students for productive careers in industry, academia, and government by providing an outstanding environment for teaching, learning, and research in the theory and applications of computing. The Department places high priority on establishing and maintaining innovative research programs to enhance its education quality and make it an important regional, national and international resource center for discovering, integrating and applying new knowledge and technologies.

The Computer Science Department aspires to be one of the top 25 departments in the nation within the next five years. Considerable effort and resources (e.g., the JSRE Initiative) are being invested to reach that goal.

Goals for Undergraduate Program

The undergraduate Computer Science program is committed to providing students with a high-quality education and prepare them for long and successful careers in industry and government. Our graduates, while eminently ready for immediate employment, will also be fully ready for focused training as required for specific positions in Computer Science and closely related areas. Our students will “learn to learn” so that they can readily adapt themselves as the field of computing changes as well as their work environment changes. Graduates interested in highly technical careers, research, and/or academia will be fully prepared to further their education in graduate school. Our graduates will have good oral and written communication skills as well as a good understanding of ethical issues related to the computing profession.

Educational Objectives for Undergraduate Program

The educational objectives of the Undergraduate Program in Computer Science are:

EO1. students should be able to apply their knowledge to the solution of practical and useful problems;

EO2. students should be able to communicate effectively and work collaboratively;

EO3. students should become successful professionals in industry, government and, if they desire, in graduate studies;

EO4. students should be able to recognize the need for lifelong learning and be able to adapt to rapid technological changes; and

EO5. students should be able to understand and deal with the ethical, societal, and global issues associated with the computing field.

Undergraduate Program Outcomes:

The Department of Computer Science at UT-Dallas offers both the B.S. Degree in Computer Science and the B.S. degree in Software Engineering. Both programs will seek ABET accreditation and most courses offered by the Department can be used in either degree plan. Since the ABET *Engineering Criteria 2000* accreditation outcomes (i.e. a-k) map well to the educational objectives above, they were adopted as outcomes for the B.S. in Computer Science. Furthermore, four additional CS-specific outcomes were added. The ABET *Engineering Criteria 2000* outcomes are that graduates must demonstrate the following:

(a) an ability to apply knowledge of mathematics, science, and engineering;
(b) an ability to design and conduct experiments as well as to analyze and interpret data;
(c) an ability to design a system, component, or process to meet desired needs;
(d) an ability to function on multidisciplinary teams;
(e) an ability to identify, formulate, and solve engineering problems;
(f) an understanding of professional and ethical responsibility;
(g) an ability to communicate effectively;
(h) the broad education necessary to understand the impact of engineering solutions in a global/societal context
(i) a recognition of the need for and ability to engage in lifelong learning;
(j) a knowledge of contemporary issues; and,
(k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

Additional CS objectives:

CS1. An ability to theoretically formulate and analyze problems in computing and information processing.
CS2. An ability to solve problems using efficient algorithms and data structures.
CS3. An ability to understand computer architecture, organization and programming languages (e.g. hardware, operating systems, networks, concepts of programming languages).
CS4. Proficiency in programming languages (e.g. Java, C/C++, assembly language)

For the purpose of assessment, the above outcomes are mapped to our program educational objectives as shown in the following table.

Our Objectives	EC2000 Outcomes
EO1	(a), (b), (c), (e), (j), (k), (cs1)-(cs4)
EO2	(d), (g)
EO3	(a) – (k), (cs1)-(cs4)
EO4	(a)-(c), (e), (h), (i), (k), (cs1)-(cs3)
EO5	(f), (h), (j)

2. Describe how your program's objectives align with your institution's mission.

Our undergraduate program goals and educational objectives are consistent with Jonsson School's mission, which in turn is consistent with UTD's mission. Education and Research in Engineering and Computer Science are emphasized in the University's mission statement. The University's strategic intent to be a nationally recognized top-tier university is embraced and expanded by the School's goal to achieve Tier-1 status within the next five years which in turn translates to achieving excellence at both the undergraduate and graduate levels for the School's existing degree programs. The goals and objectives at the departmental level are fully supported upward at the School and University levels.

Note: On the following page is a table which can be filled out with pertinent information relating to objectives, their measurement, and their effect on the implementation of program improvements.
Implementation of Objectives. Please complete the following table.

Table 1: Implementation of Objectives.

The table below shows the result of improvements identified and subsequent actions taken as a result of in-class assessment.

Objective	How Measured	When Measured	Improvements Identified	Improvements Implemented
EO1	In-class assessment	Each semester	Software design ability	1. Switch from C++ to Java and introduce object-oriented design from the start (Fall 2003); 2. Redesign the beginning programming course sequence to assure uniform, better background and enable better coverage (Fall 2004); 3. More design projects in various classes (Fall 2004); 4. Students must pass both the theory portion of each course as well as the programming portion (Spring 2005).
	Surveys	Each Year		
			Update curriculum (coverage of “new” areas)	Adjust coverage of recent developments in areas covered by existing classes (ongoing); Addition of several electives, for example, CS 4393 on computer and network security (Fall 2004)
			More Hardware Design	1. Replaced CS 2325 which was mostly Assembly language programming with CS 2310 - Introduction to Digital Systems (Fall 2003)
EO2	In-class assessment	Each semester	Professional oral communication skills, and contemporary issues	1. Redesign of ECS 3390 (ongoing); 2. More project reports, presentations in various classes (ongoing).
	Surveys	Each year		
EO3	Surveys (including job market)	Each year	1. Facilitate transition to graduate studies (more demand for MS). 2. Facilitate transition to industrial jobs	1. Introduction of fast-track MS program. 2. Require each (tenure-track) faculty to teach at least one undergraduate class per year (Fall 2004).
	Industrial Advisory Board	Each year		1 Expansion of co-op program (ongoing). 2. Require each (tenure-track) faculty to teach at least one undergraduate class per year (Fall 2004).

Objective	How Measured	When Measured	Improvements Identified	Improvements Implemented
EO4	In-class assessment	Each semester	Learn by doing.	1. More design projects (ongoing); new electives (Fall 2004); 2. update existing classes (ongoing)
	Surveys	Each year		
	Industrial Advisory Board	Each year		
EO5	In-class assessment	Each semester	Coverage of ethics, social issues.	1. Incorporate in existing classes (e.g. CS I, II, CS 3354 – ongoing); 2. Addition of elective CS 3385. 3. Made ISSS 3360--Politics and Values in Business and Technology -- a requirement.
	Surveys	Each year		
	Industrial Advisory Board	Each year		

Standard I-3. Data relative to the objectives must be routinely collected and documented, and used in program assessments.

See I-5. C

Standard I-4. The extent to which each program objective is being met must be periodically assessed.

See I-5.C

Standard I-5. The results of the program's periodic assessment must be used to help identify opportunities for program improvement.

C. Assessments. Describe your procedure for periodically assessing the extent to which each of the above objectives is being met by your program.

Include:

- frequency and timing of assessments
- what data are collected
- (should include information on initial student placement and subsequent professional development)
- how data are collected
- from whom data are collected
- (should include students and computing professionals)
- how assessment results are used and by whom

Attach copies of the actual documentation that was generated by your data collection and assessment process since the last CSAC visit (or for the past three years if this is the first visit). Include survey instruments, data summaries, analysis results, etc.

Overview

In an effort to evaluate the institutional effectiveness of our programs, the Erik Jonsson School of Engineering and Computer Science established the Office of Assessment in 2001 to meet the needs of various programs. This office provides information and support to the administrative and academic units in a broad range of activities. Currently, the Office of Assessment implements the use of the Undergraduate Level Senior Exit Survey , the Alumni Focus Group Meeting and Survey, and the Employer Focus Group and Survey. The Undergraduate Level Senior Exit Survey is collected every semester and sent for data analysis to EBI, Inc., every year. The Alumni Survey is conducted every two years and sent for data analysis to EBI, Inc., while the Employer Survey is done every few years or so to gauge changes in the industries that employ our graduates (Note that the Employer Survey is done sporadically because our annual Industrial Advisory Board meetings serves as a useful conduit of information from the industry).
(Reference: Office of Assessment http://www.utdallas.edu/dept/eecs/ouga/assessment.html)

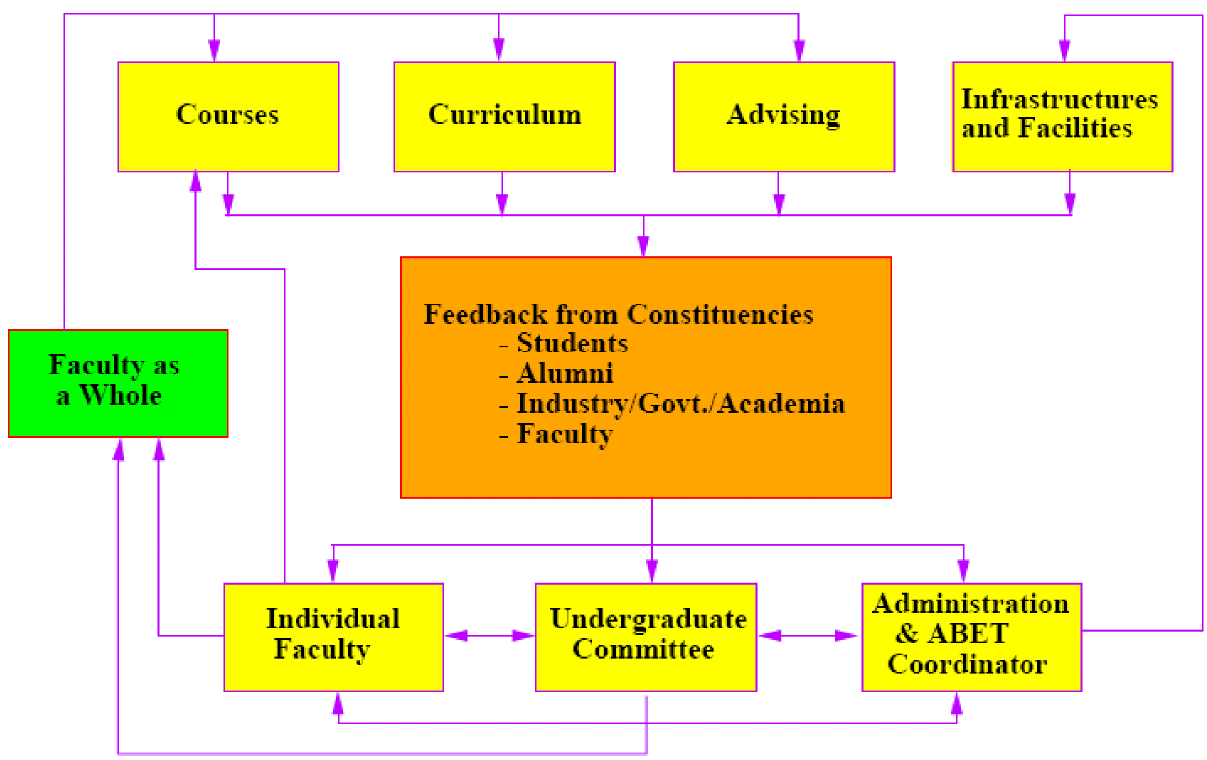


Figure I-5.1. Computer Science Program Assessment Flowchart

An assessment plan has been developed to ensure that graduates have achieved the Program Outcomes. The general assessment process flow chart is shown in Figure I-5.1. The curriculum committee (in collaboration with the ABET coordinator) is at the heart of this assessment and improvement process. They monitor the surveys from the students, alumni and industry, and look for areas where improvement is needed. They also solicit input from the faculty and our constituencies (e.g., industry, alumni, etc.) on a regular basis to determine if other changes are needed in the program. The ABET coordinator makes presentations at faculty meetings to keep the faculty up-to-date on assessment issues.

Additionally, the Office of Assessment collects informal input that is provided to the curriculum committee. The three area committees within the department also provide suggested changes to the curriculum committee. These three area committees provide advice on curricular matters related to Software Engineering (SE) courses, Telecom Engineering (TE) courses, and Computer Science courses. Using these inputs, changes are identified and recommendations for their implementation made. The recommendations made by the curriculum committee are brought to the faculty for further discussion, vote and adoption.

In order to obtain statistically reliable assessment data, several assessment devices were used:

- Pre-graduation
    - **a.** In-class assessment (by all faculty)
    - **b.** Senior exit surveys
    - **c.** Student's transcripts and evaluations

- Post-graduation
    - **a.** Alumni surveys
    - **b.** Employer surveys
    - **c.** Job placement data, and admission data to graduate schools

The main assessment devices used to ensure that students achieved the Program Outcomes were: (1) in-class assessment of class learning objectives; these were done for almost all core classes in the CS program since Fall 2002, and all classes (including electives) since Fall 2004 (2) senior exit survey, and (3) alumni survey. A detailed explanation of how the above assessment devices were used and implemented follows.

In-class assessment

For each undergraduate core course, course learning objectives (CLO) were developed. Each CLO is mapped to one or more of the *a-k* ABET outcomes and the additional CS outcomes (cs1-cs4). Before each semester, course instructors review the educational objectives and suggest revisions as needed. Minor revisions are adopted with the approval of the ABET coordinator; more significant changes or suggested revisions for which there is no consensus among the instructors and the ABET coordinator are referred to the curriculum committee.

At the end of each semester, course instructors are required to rate the student's ability in achieving the course learning objectives on a scale of 1 to 5, with 5 being exceptional and 1 being poor. In addition, instructors are required to: (1) list the class materials (e.g., homeworks, exams, projects) used for assessing each CLO, and (2) the criteria used in obtaining the rating scores (see Example in Table 2). The Office of Assessment is responsible for collecting all the CLO forms each semester and distributing the rating scores to the departments. Each core class has a course coordinator who analyzes the data for various sections of that course offered during one semester and across different semesters. The analysis and inferences drawn from the analysis is submitted by each coordinator to the departmental curriculum committee. Current course coordinators are shown in the Table below.

Course	Course Coordinator
CS1337	**Steinhorst**
CS2305	**Farage**
CS2310	**Harrison**
CS2336	**King**
SE3306	**Cooper**
CS3305	**Van Ness**
CS/SE 3341	**Amman**
CS3345	**Uma, Bereg**
CS3354	**Chung**
ECS3390	**Faler-Sweany**
CS4340	**Dattatreya**
CS4348	**Ozbirn**
CS4349	**Daescu**
SE4351	**Sullivan**
SE4352	**Chung**
SE4367	**Cangussu**
SE4381	**Leubitz**
CS4384	**Huynh**
SE4485	**Russo**
CS4390	**Cobb**

The ABET coordinator (currently Associate Head, Dr. Gopal Gupta) in collaboration with the undergraduate curriculum committee and the course coordinators also analyzes the data and computes the mean rating scores of the a-k ABET outcomes across all courses. The mean rating scores of the *a-k* ABET outcomes are compared against a threshold, and if the scores fall below that threshold, that signifies that a change needs to be made in the curriculum or program. The threshold level was initially set to 3.5 (out of a maximum score of 5) for all ABET outcomes. Before any action is taken, possible changes to the curriculum are discussed in faculty meetings. Feedback from the industrial advisory board is also sought. This process ensures that the loop is closed for program improvement. One of the main advantages of in-class assessment is that *all* faculty are involved in the assessment process and consequently in program improvement. A number of improvements have been suggested and implemented based on this assessment. These are summarized in Section I-6.

Table I-5.2 below shows an example CLO form for the Operating Systems course (CS 4348-Fall 2003).

Assessment of CS4348/501 (Operating Systems Concepts)

Instructor: Greg Ozbirn Semester: Fall 2003

On a scale of 1 to 5, with 5 being exceptional and 1 being poor, rate the student's:

Class learning objectives	Rating	ABET objective	Add. CS objective	Material Used	Criteria Used
Understand modern hardware organizations and their impact on OS design	3.42	c	3	Midterm q4	A =5; B=4; C=3; D=2; F=1
Ability to understand the process state model	4.55	c	3	Midterm q1	
Ability to understand process description, and process control	4.35	c	3	Midterm q3	
Ability to understand the difference between processes and threads	3.72	c,k		Midterm q2	
Ability to understand thread state, thread synchronization, ULTs vs KLTs	2.84	c,k	4	Midterm q5	
Ability to understand long term, medium term, and short term scheduling	4.28	c		Final q3a (p3)	
Ability to understand and evaluate short-term scheduling algorithms	4.24	a,c,e,k	1,2	Final q3b (p3), q3 (p6)	50/50
Ability to understand software vs hardware approaches to mutual exclusion	3.51	c	3	Midterm q6	
Ability to use semphores for mutual exclusion and synchronization	3.05	c		Midterm q8	
Ability to analyze concurrent programs	4.45	a,c,e,k		Project 1	
Ability to analyze concurrent programs in terms of deadlock and starvation	1.70	a,c,e,k	1,2	Midterm q7	
Ability to understand strategies for deadlock: prevention, avoidance, detection	4.40	a,c	1,2	Final q1 (p2)	
Ability to understand the requirements of memory management	4.36	c	3	Final q2a (p2)	
Ability to understand simple memory partitioning schemes	4.78	c		Final q5,8 (p1)	50/50
Ability to understand virtual memory systems, especially page or segmentation	4.72	c,e	3	Final q4,9 (p1), q2a,b,c (p2)	q4,9 = 50, q2a,b,c=50
Ability to understand the benefits of caching in virtual memory management	3.22	a		Final q2b (p2)	
Ability to perform a comparison of page replacement algorithms	3.98	a,k	1,2	Final q2 (p5)	
Ability to understand resident set management, and the working set policy	2.51	c,e		Final q2c (p2)	
Ability to understand programmed vs. interrupt-driven I/O, DMA	4.69	a.c	3	Final q4b (p3)	
Ability to understand I/O buffering policies and disk scheduling policies	3.68	a		Final q4c (p3), q5a (p4)	50/50
Ability to understand disk arrays & caching	4.02	c,k	3	Final q5b,c (p4)	50/50
Ability to understand file system organization and access	1.00	a,e,k	3	Final q4a (p3)	

Table I-5.2. Example assessment form used for evaluation of Operating Systems (CS 4348).

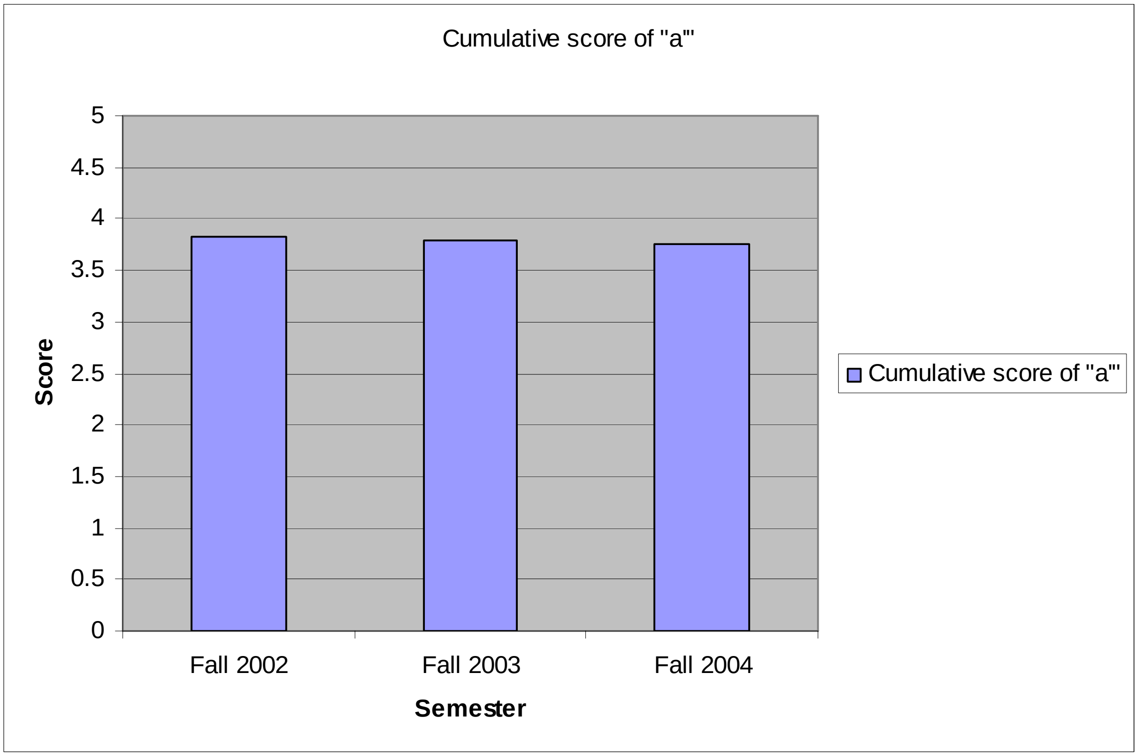
Course	a	b	c	d	e	f	g	h	i	j	k	cs1	cs2	cs3	cs4
CS1337	×		×		×						×				×

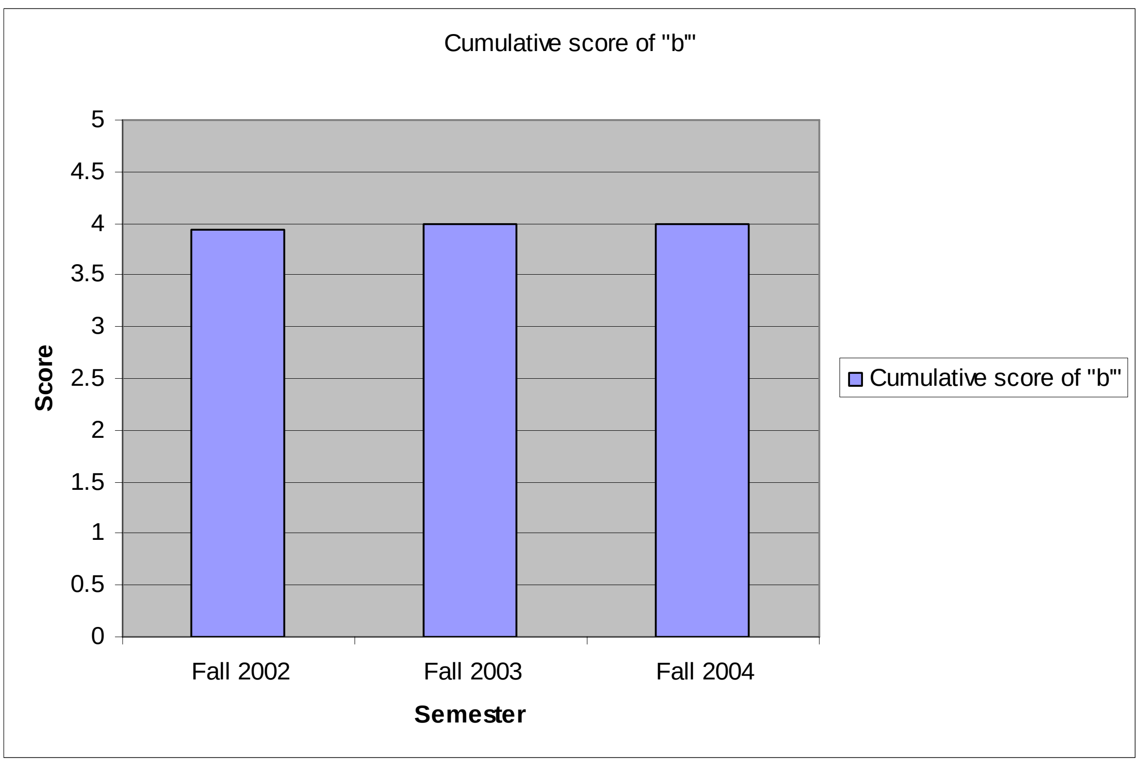
CS 2110	×	×	×								×	×			
CS2305	×				×										
CS2310	×	×	×		×						×	×			×
CS2336	×		×		×						×	×	×		×
CS3305	×				×										
CS 3341	×	×			×						×	×			
CS3345	×		×		×						×	×	×		×
CS3354	×		×	×	×	×	×	×			×	×	×		×
ECS3390	×			×		×	×		×	×	×				
CS 4337	×										×	×	×	×	×
CS4340	×		×		×						×	×		×	
CS4348	×		×		×						×	×	×	×	×
CS4349	×		×		×						×	×	×		
CS4384	×		×		×						×	×			
CS4390	×		×		×						×	×	×	×	×
ISSS3360						×			×	×					

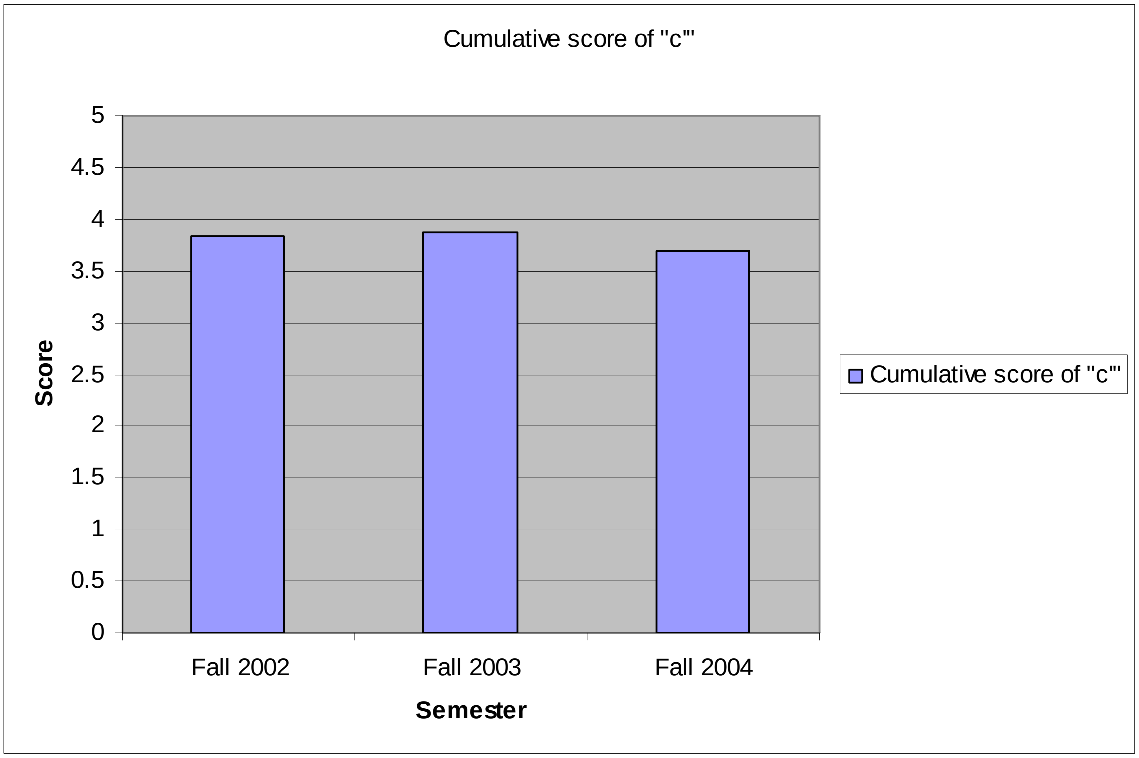
Table I-5.3. Mapping of course learning objectives to outcomes.

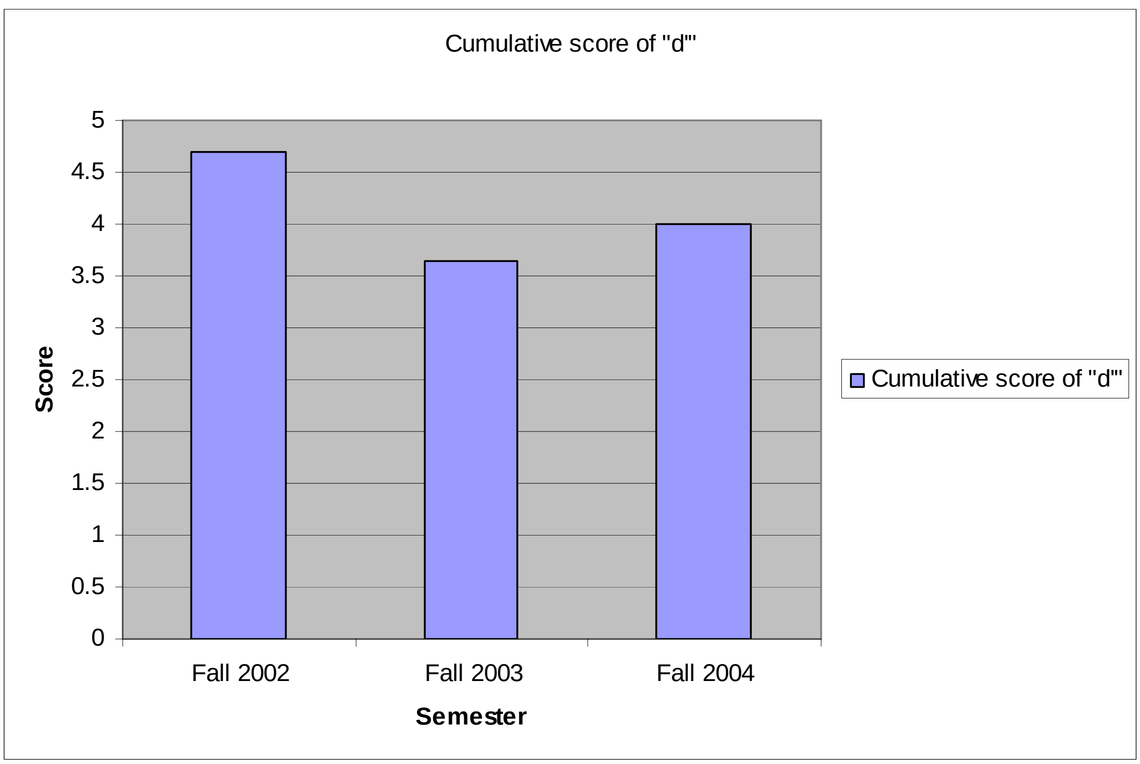
Table I-5.3 shows the mapping of undergraduate CLOs to outcomes. For Computer Science core courses most of the a-k and cs1-cs4 outcomes are evaluated by multiple courses, thereby strengthening the reliability of our results. Six outcomes (b, d, f, h, i, j) are evaluated by relatively fewer courses. However these outcomes are also evaluated in the alumni and senior exit surveys as well as by the feedback we get from our Industrial Advisory Board and Employer Surveys.

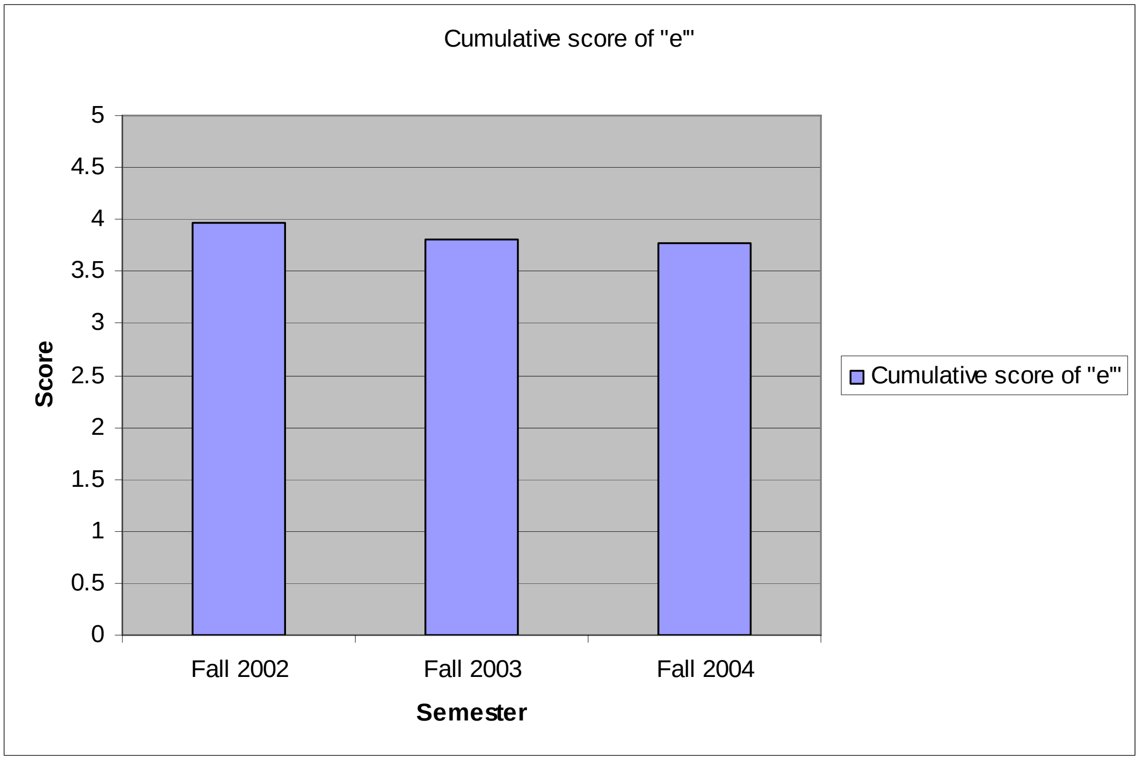
The next figure shows the mean rating scores of CS core classes for Fall 2002, 2003 and 2004 semesters. We have chosen a one year interval since any changes made to the program or curriculum can take up to a semester before their effect starts showing up in outcomes scores. Note that for outcomes *a*, *b*, *c*, *e*, and *k*, that are directly related to learning, applying technical skills and problem solving, the cumulative scores are more or less the same (but in all cases they are at or above 3.5). For categories *f*, *g*, *h*, and *j*, which deal with broadening aspects, the improvements can be seen from the bar graphs. For category *d*, which relates to ability to work in multidisciplinary teams, the data shows a downward trend. Certainly our students do not get enough experience working in teams, and we are working to rectify this by adding more team oriented projects in our curriculum. For CS1-CS4 criteria, all the scores are above the 3.5 threshold, however, there is a slight drop in Fall 2004 semester. The only explanation we can find is that perhaps more tenure-track faculty members taught undergraduate classes in the Fall, and perhaps they are more conservative judges of student performances.

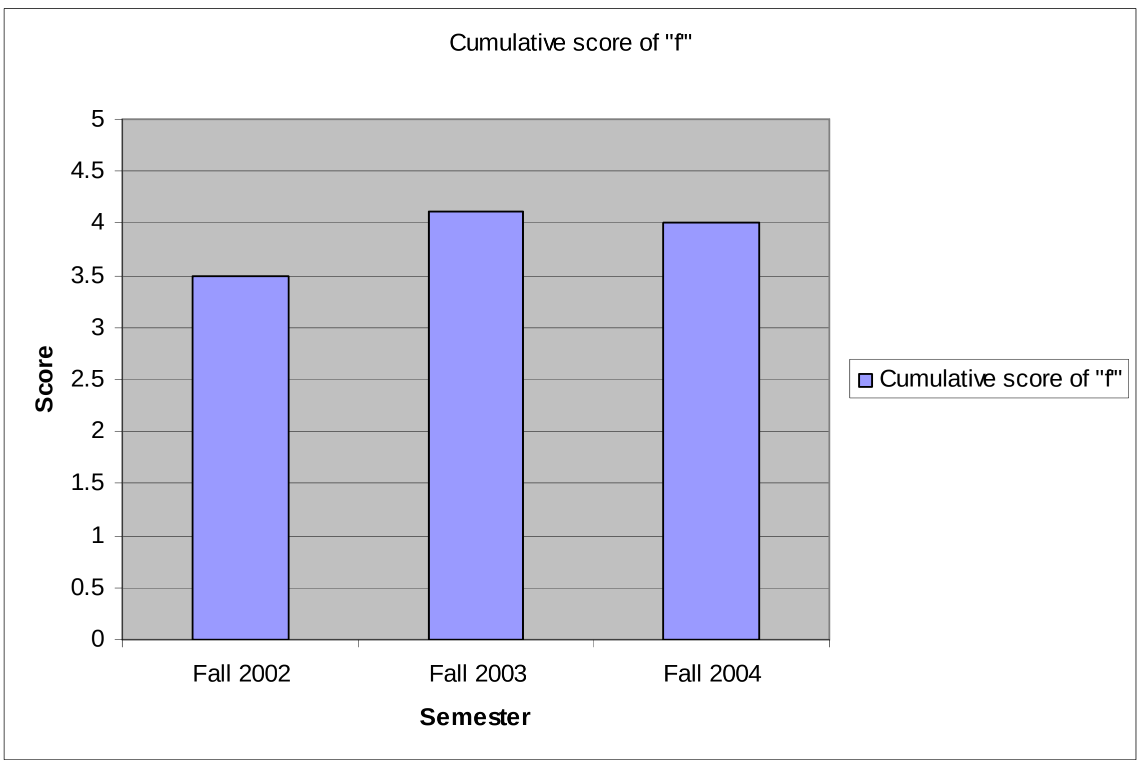

Cumulative score of "a"'
Score
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "a"'

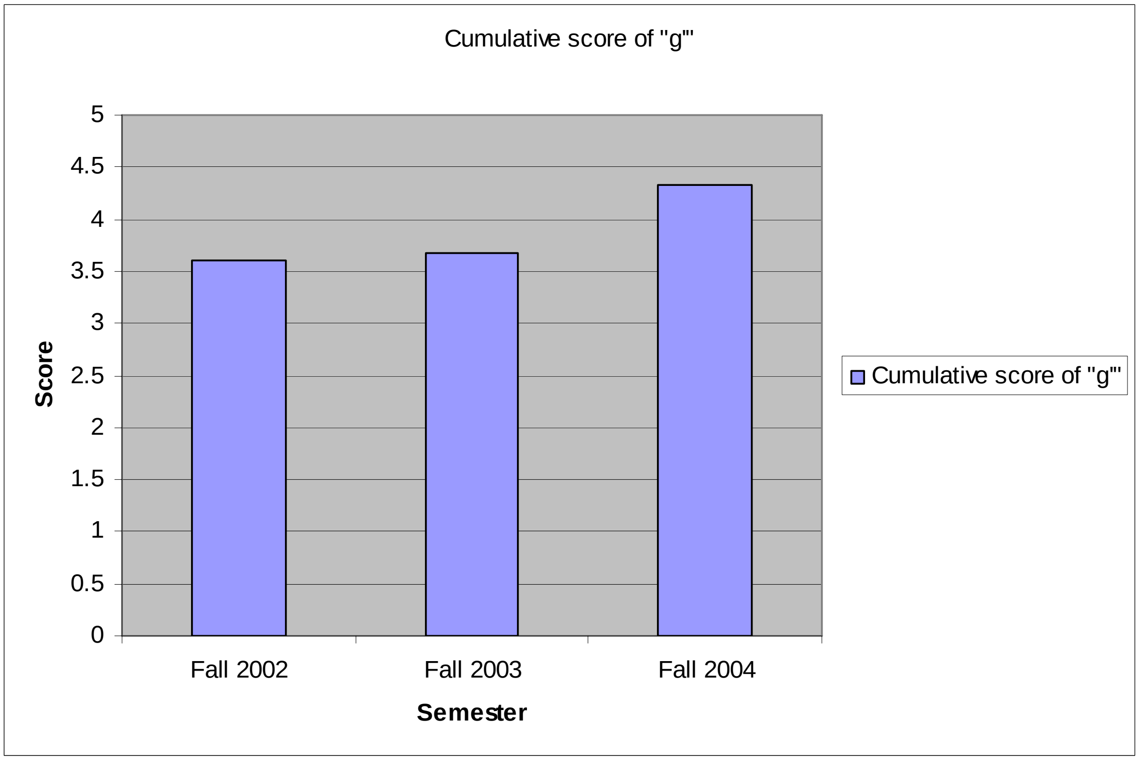

Cumulative score of "b"'
Score
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "b"'


Cumulative score of "c'"
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "c'"

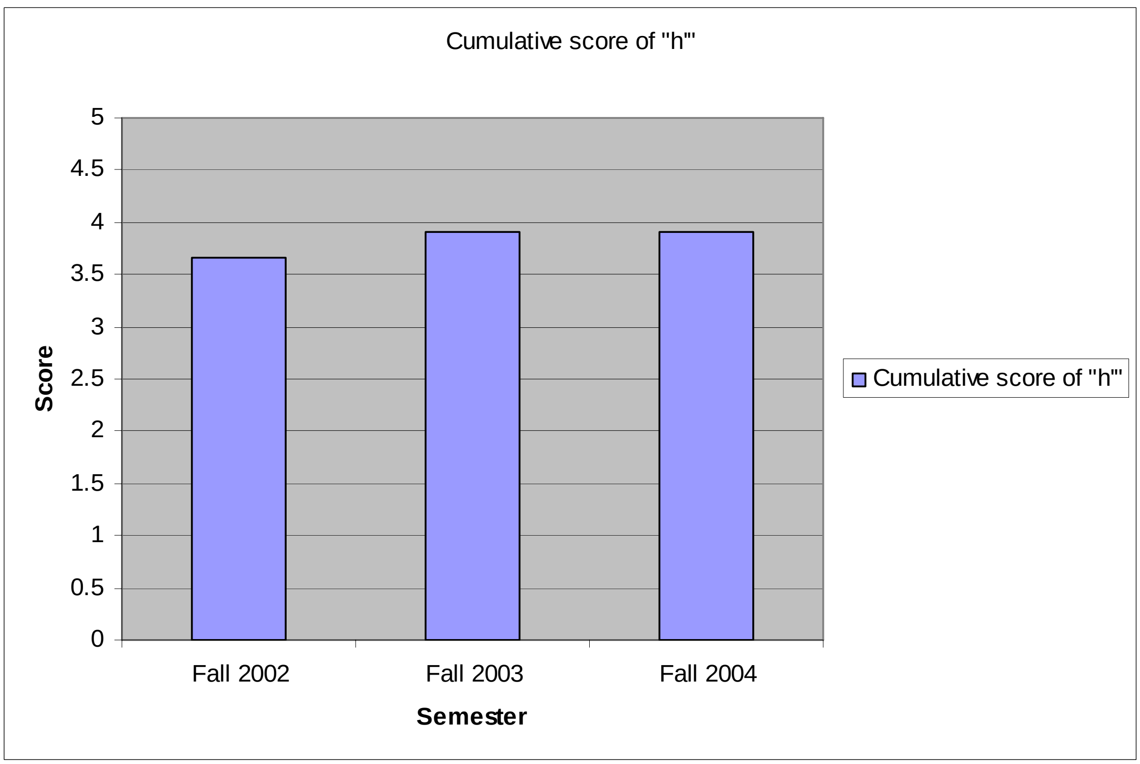

Cumulative score of "d'"
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "d'"

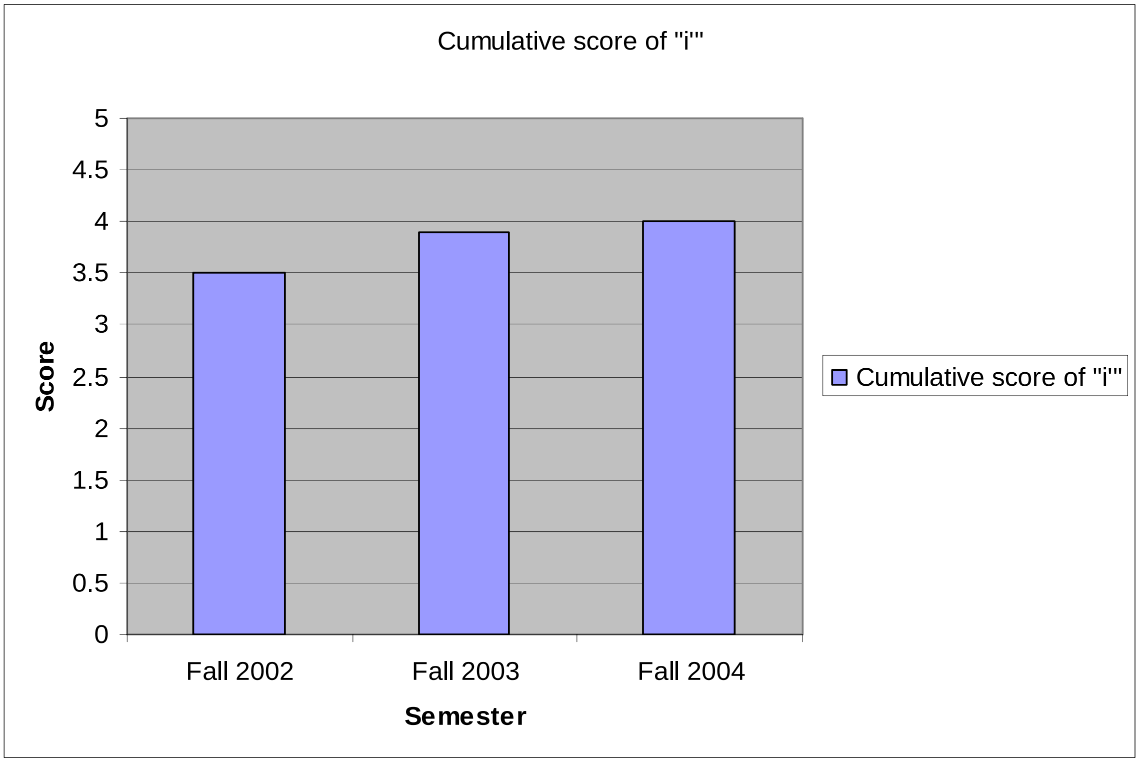

Cumulative score of "e"'
Score
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "e"'

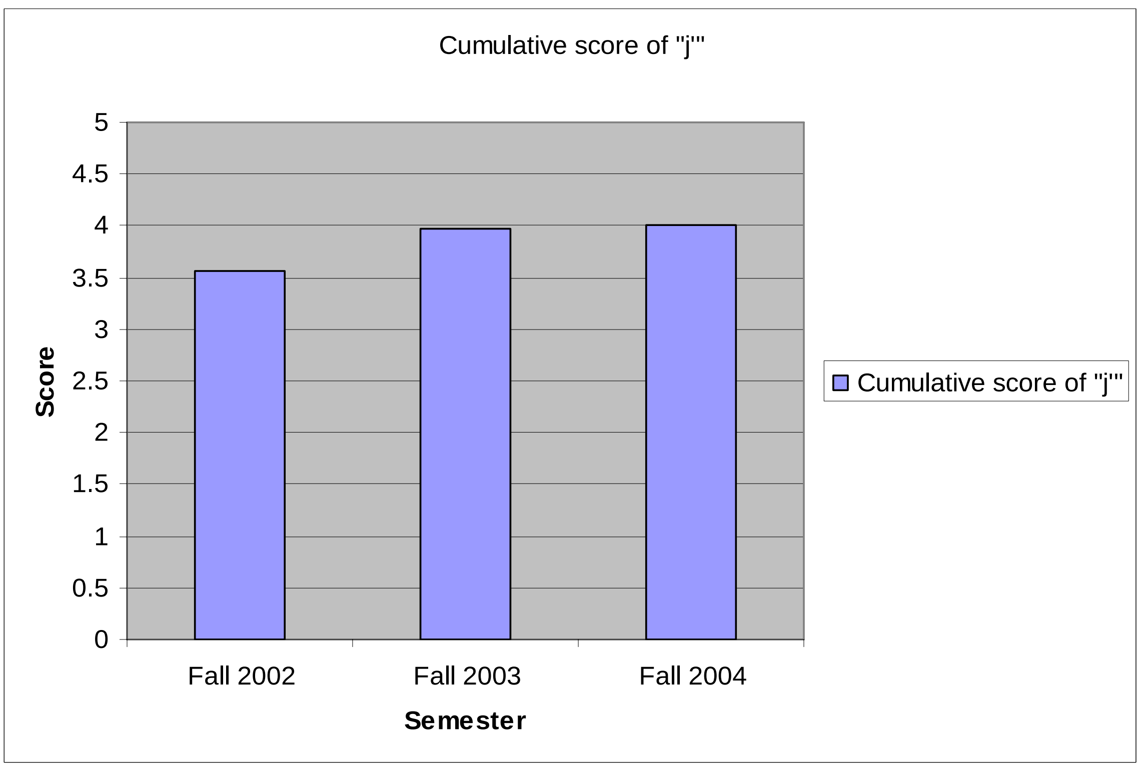

Cumulative score of "f"'
Score
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "f"'

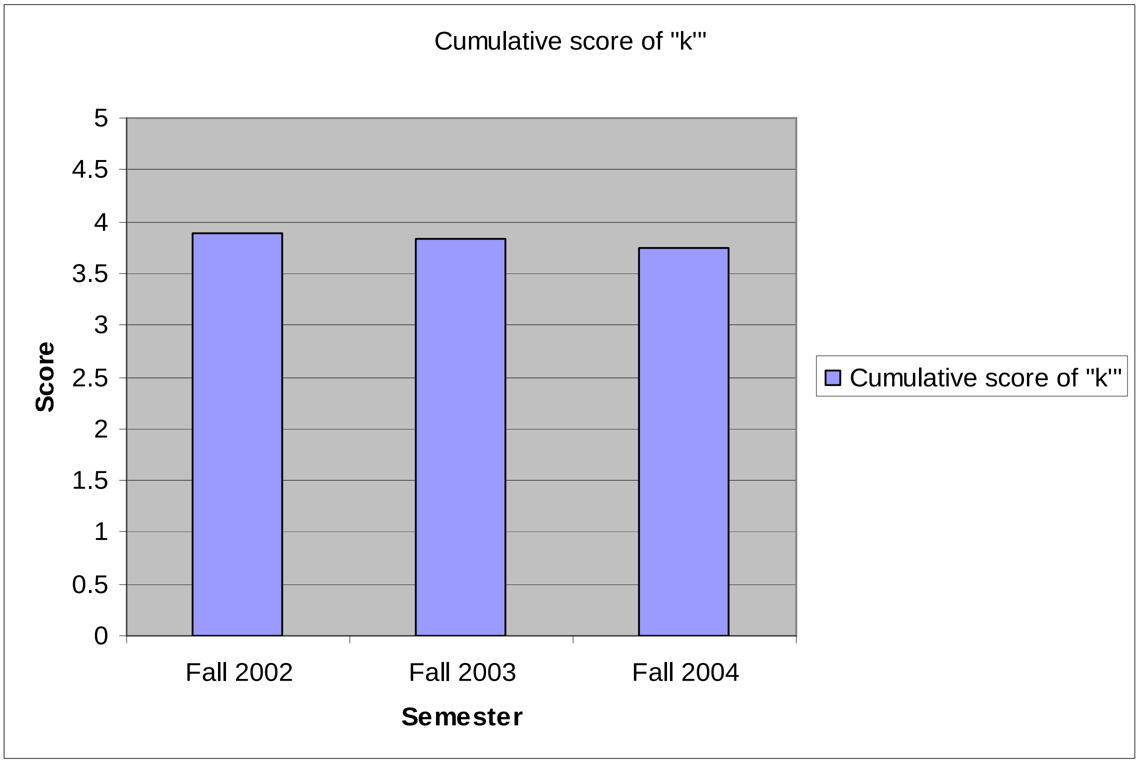

Cumulative score of "g"'
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "g"'

.

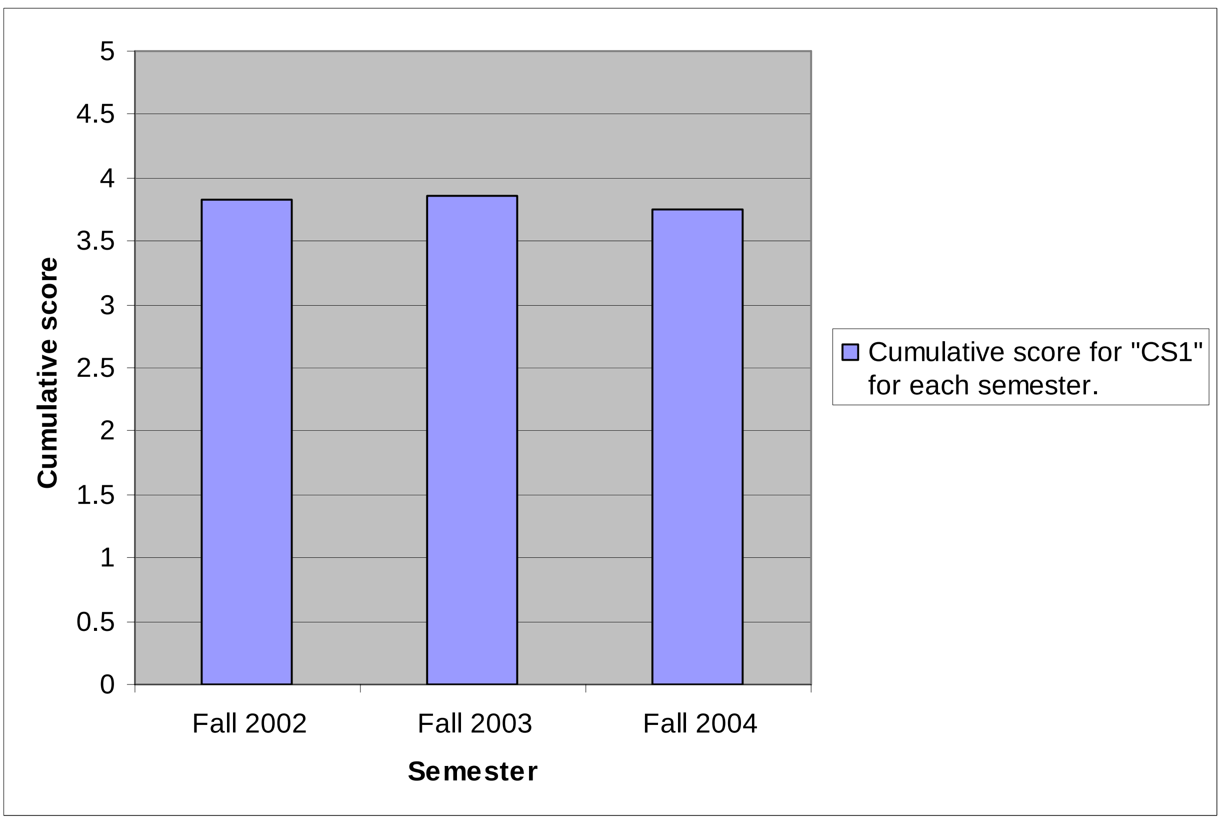

Cumulative score of "h"'
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "h"'


Cumulative score of "i'"
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "i'"

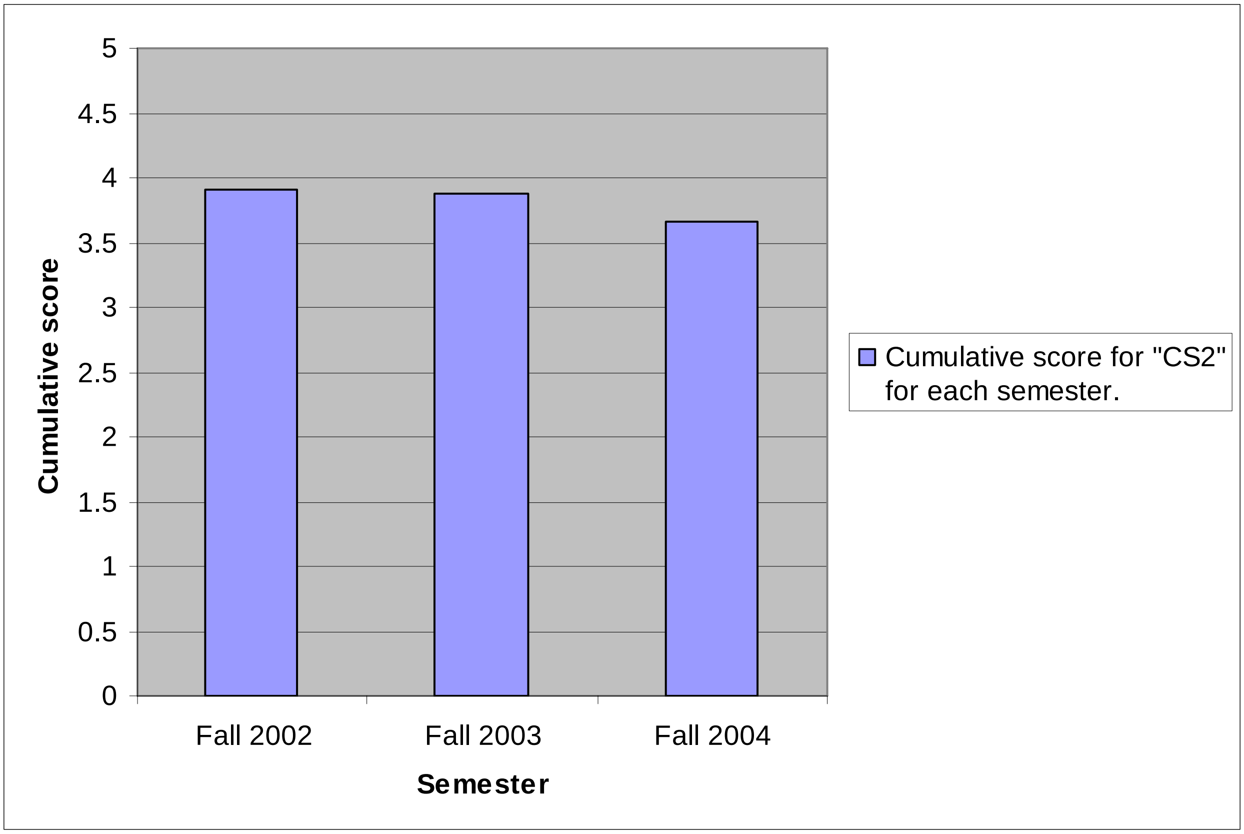

Cumulative score of "j'"
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "j'"

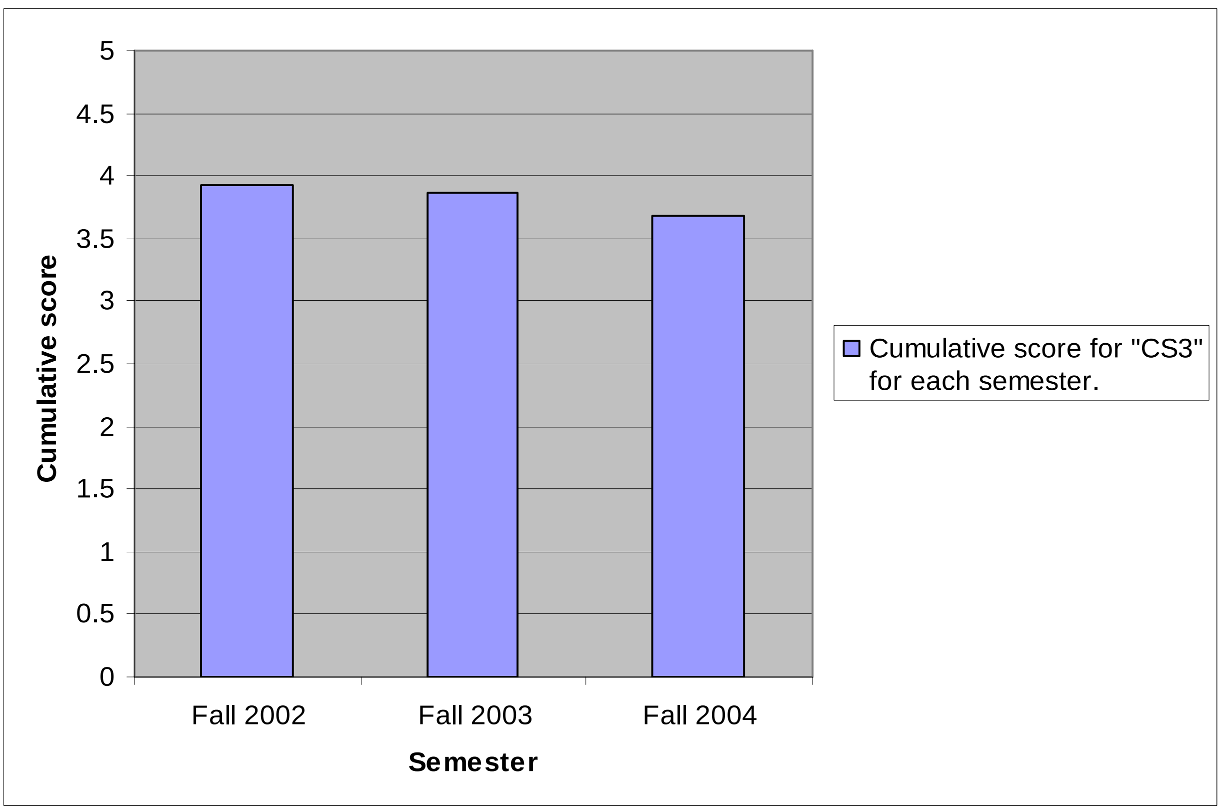


Cumulative score of "k'"
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score of "k'"



5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Cumulative score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score for "CS1" for each semester.


5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Cumulative score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score for "CS2" for each semester.


5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Cumulative score
Fall 2002
Fall 2003
Fall 2004
Semester
Cumulative score for "CS3" for each semester.

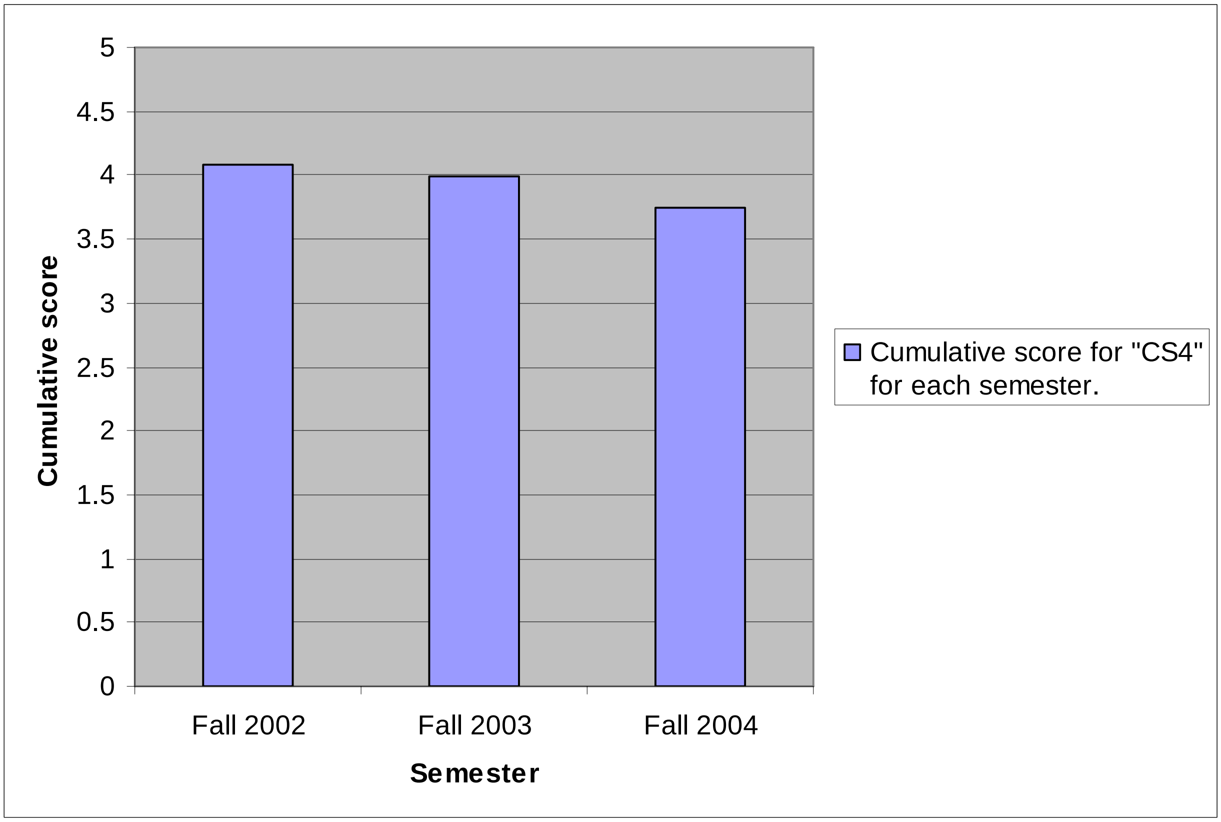


Figure I-5.4. Mean a-k and CS1-CS4 rating scores for in-class assessment of CS core courses for Fall 2002, Spring 2003, Fall 2003

Note that in-class assessment for Computer Science courses was initiated in Fall 2002 following the decision to seek ABET accreditation for both the B.S. in Computer Science and the B.S. in Software Engineering degrees (in April 2002). The process was adjusted every semester in order to reduce inconsistencies, improve reliability of the data. Parallel to this, the ABET accreditation process and the role of in-class assessment in it were discussed with instructors at least once a semester. In Spring 2004, the process was reviewed again and further adjustments were made to improve consistency across the curriculum. The most recent set of guidelines to the instructors is attached as Appendix VII. The assessment data for Spring and Summer 2005 is currently being compiled.

Teaching Evaluations

Approximately three weeks before the end of the semester, a 15-minute period at the beginning of each class is reserved for student evaluations, by prior arrangement with the instructor. The evaluation is conducted by a Teaching Assistant who has no formal relationship with the instructor. The instructor is not present during the evaluation. At the beginning of the 15-minute evaluation period, students are asked to complete course evaluation forms that detail the efficacy of the instructor, the TA (if applicable), and course materials. The forms and standards of the student course evaluations are customized by the Committee on Effective

Teaching (CET) of the School of Engineering and Computer Science. This committee is appointed by the Dean to ensure that the University's policy of effective teaching is met as stated in University Policy Memorandum 96-III.21-70. The results of these evaluations are made available to both the individual instructors and the Department or Program Head.

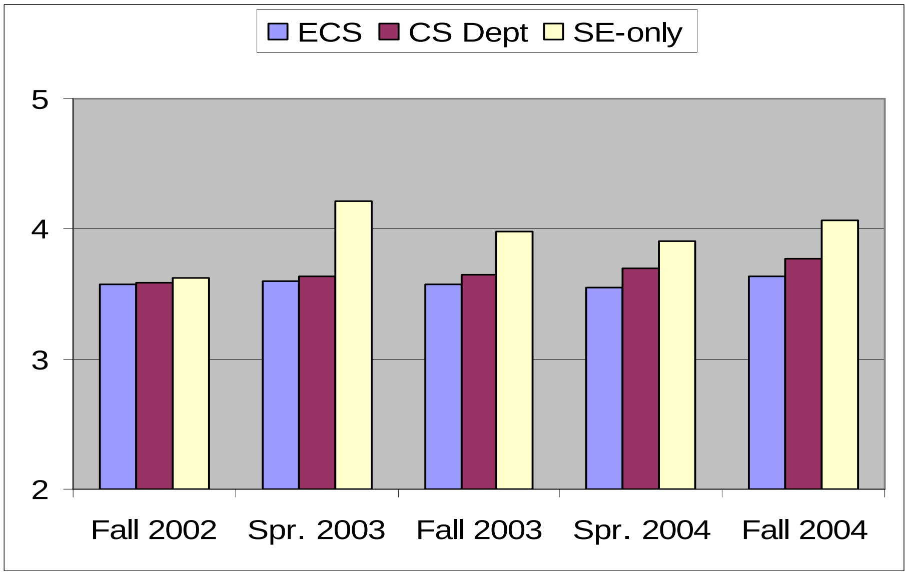


Figure I-5.5: Teaching evaluation ratings comparison

Figure I-5.5 shows the average teaching evaluation score for selected questions identified by the office of Assesment as reflecting overall quality of instruction (9 questions). The ratings for three different groups of classes for each long semester from Fall 2002 to Fall 2004 are shown. The first group includes all undergraduate classes taught in the School of Engineering and Computer Science; the second group reflects undergraduate classes taught by the Department of Computer Science ; the third group are six classes that have the SE prefix only – they are major requirements only in the BS-SE degree and are taken mostly by Software Engineering majors. The comparison indicates that students rate the SE only classes higher than the School, Department averages.

The teaching evaluation forms also request student comments and those are often more useful than the ratings themselves. A review of the comments suggests that the main reason for the higher ratings for the SE-only group of classes may be the increased emphasis on projects related to practical software development in the SE-only group. The comments often are useful in identifying specific problems with a class or instructor. One example is a high concentration of complaints about the overlap between the content of SE 3306 and CS 3305 in Spring 2004. The instructor took those to heart, provided more material outside the textbook and that complaint was completely absent in the Fall 2004 evaluations.

Senior Exit Surveys

The first formal exit survey for graduating seniors from the CS Department was introduced and implemented in Fall 2000. This survey was developed by the Office of Assessment as a pilot study; the results were distributed to the department in May 2002.

Since then, ECS has acquired a survey instrument from a private vendor (EBI, Inc.) and the Office of Assessment continues to implement this process every semester. The results of this survey, in quantitative form, have been distributed to the department each year. Collection of this data has been largely successful with return rates of 85% in 2001, 88% in 2002, 98% in 2003 and 66% in 2004. In addition, we have collected the written comments from the surveys; this qualitative portion of the data is distributed together with the quantitative portion.

The Senior Exit Survey asked students several questions that focused on the ABET outcomes. These questions are as follows:

Q38. To what degree did your engineering education enhance your ability to design experiments?

Q39. To what degree did your engineering education enhance your ability to conduct experiments?

Q40. To what degree did your engineering education enhance your ability to analyze and interpret data?

Q42. To what degree did your engineering education enhance your ability to function on multidisciplinary teams?

Q45. To what degree did your engineering education enhance your ability to understand the impact of engineering solutions in a global/societal context?

Q46. To what degree did your engineering education enhance your ability to use modern engineering tools?

Q47. To what degree did your engineering education enhance your ability to communicate using oral progress reports?

Q48. To what degree did your engineering education enhance your ability to communicate using written progress reports?

Q49. To what degree did your engineering education enhance your ability to pilot test a component prior to implementation?

Q51. To what degree did your engineering education enhance your ability to recognize the need to engage in lifelong learning?

Q52. To what degree did your engineering education enhance your ability to apply your knowledge of mathematics?

Q53. To what degree did your engineering education enhance your ability to apply your knowledge of science?

Q54. To what degree did your engineering education enhance your ability to apply your knowledge of engineering?

Q55. To what degree did your engineering education enhance your ability to identify engineering problems?

Q56. To what degree did your engineering education enhance your ability to formulate engineering problems?

Q57. To what degree did your engineering education enhance your ability to understand contemporary issues?

Q58. To what degree did your major design experience build on the knowledge of previous course work?

Q59. To what degree did your major design experience build on skills from previous course work?

Q60. To what degree did your major design experience incorporate engineering standards?

Q65. To what degree did your major design experience address ethical issues?

ABET outcome	Rating computed from EBI Exit Survey Questions
A	(Q52+Q53+Q54)/3
B	(Q38+Q39+Q40)/3
C	(Q49+Q58+Q59+Q60)/4
D	Q42
E	(Q55+Q56)/2
F	Q65
G	(Q47+Q48)/2
H	Q45
I	Q51
J	Q57
K	Q46

Table I-5.6: Mapping between a-k outcomes and Exit Survey Questions

The mapping of the Program Outcomes into EBI survey questions is shown in Table I-5.6. Figure 3 shows the results for 2002, 2003, 2004 graduates (academic year). The mean scores for 2002, 2003 are for CS, SE graduates as a single group (no distinction was made in the survey instrument). The two programs were individually identified in 2004 and the results for both are shown in Figure I-5.7. There is an obvious dip in the ratings for CS/SE in 2003 – perhaps reflecting the general concern with the job market. Ratings for 2004 are back up perhaps reflecting increasing optimism with BS-SE graduates generally feeling better about their studies than their peers receiving the BS-CS degree. The connection to the job market seems to be too obvious to dismiss in favor of programmatic reasons for the ratings dip in 2003. The range across the a-k outcomes is relatively small.

Overall, the scores were comparable to the scores obtained by in-class assessment (Figure I-5.9) of the class learning objectives. The scores obtained for some outcomes were higher than the corresponding scores obtained by in-class assessment (Figure I-5.9). This suggests that our students believe that the existing program does a better job in achieving all the outcomes than suggested by in-class assessment

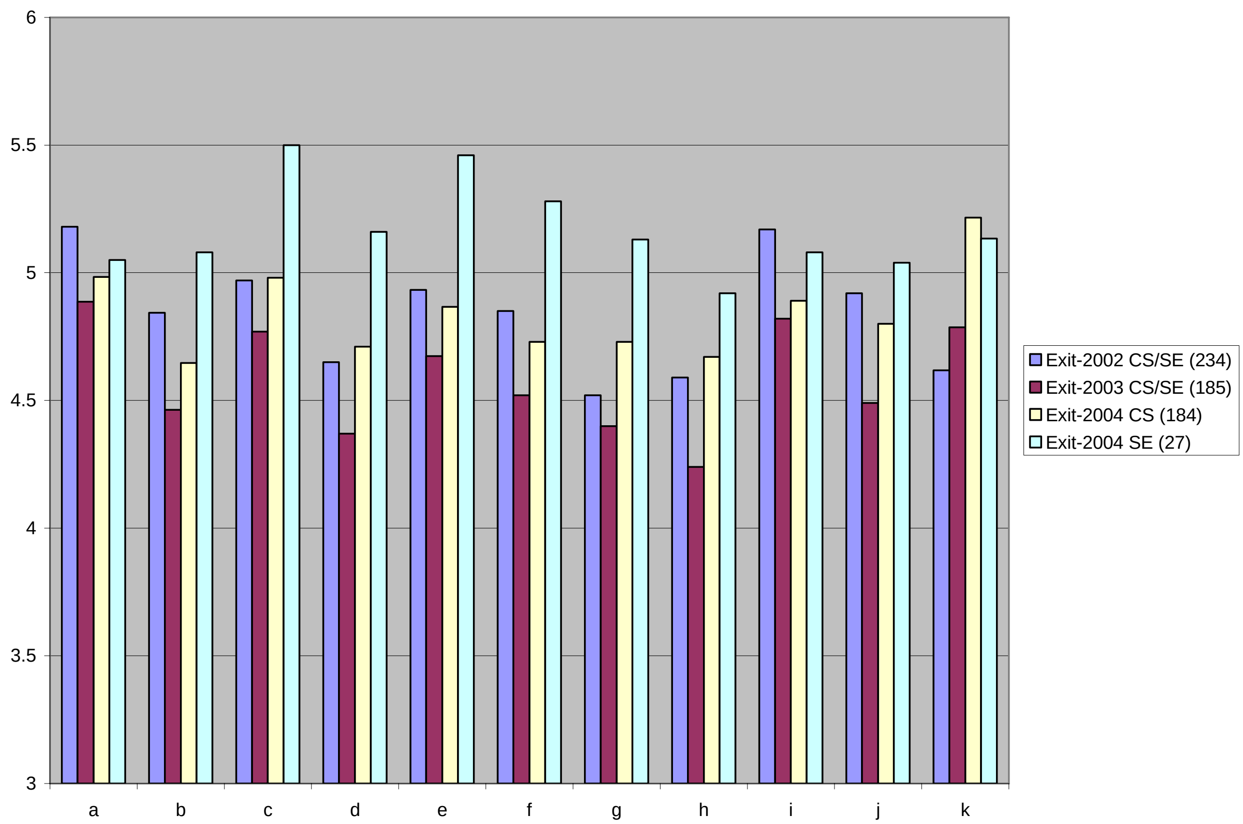


Figure I-5.7. Senior exit survey results obtained for 2002-2004.

The exit surveys also provide a comparison to 6 other universities selected as “peers” and with a larger group of Carnegie class institutions that use EBI and the group of all universities that use EBI. The comparisons place UTD graduates near the bottom within each group in almost all measures. The telling statistic is that the ratings for all schools are in a fairly narrow band. This can be viewed as indication that we are doing well since our ratings are close to those of well known schools. It can be viewed as indicative of major problems with our programs since we rank near the bottom in most categories. It can be viewed as non-applicable since our School consists of only two departments and 4 degree programs at the undergraduate level while typical Engineering schools will have many more departments and degree programs , many of them with small sizes (usually resulting in higher ratings). The validity of the comparison is also questionable since students rate their own experience at their own university on a common scale but usually have little opportunity to evaluate their experience at one University vs. what their experience would be at another. The approach we took is to look at the data but to not base any programmatic changes on them.

Alumni and Employer surveys

The Alumni Survey is conducted every other year. It is spearheaded by a focus group meeting that is followed by distribution of the survey instrument. This process allows us to collect both quantitative and qualitative data. The data is analyzed by EBI, Inc

The first focus group meeting was held in Fall 2001. Quantitative portions of the results were collected from the paper-and-pencil surveys during the fall 2001, spring 2002, and summer 2002 semesters. The qualitative data portion is a combination of focus group results from fall 2001 and written comments on our survey forms from fall 2001, spring 2002, and summer 2002. Summary reports were distributed in January, 2003. Subsequent alumni survey was conducted in 2003-2004. As done in the past, the alumni survey portion implementation was processed during Spring-Summer 2004 semesters and the results distributed in early October 2004. Currently, the Office of Assessment is implementing alumni survey for 2004-2005. As done in the past, our alumni focus group meeting was held in fall 2004 semester.

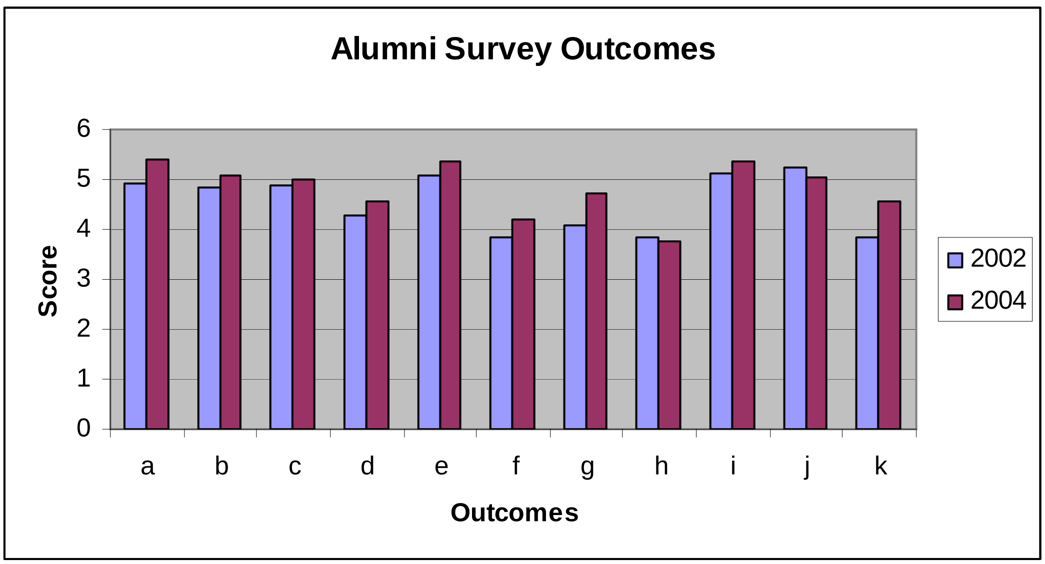


Figure I-5.8. Alumni survey results for 2002 & 2004.

Driven by the need to collect quantitative data for evaluating the *a-k* ABET outcomes, we adopted the EBI alumni surveys for 2002 and 2004.The initial return rate for the year 2002 was disappointingly low (only 8 alumni), however, it improved for 2004 (44 responses). Figure I-5.8 shows the mean rating scores for outcomes *a-k* for years 2002 and 2004. The ratings were computed from the EBI Alumni survey. The mapping between ABET outcomes and EBI alumni survey questions is shown below in Table I-5.9 (the number of responses received for each question were taken into account for outcomes *a, b, e* and *g* since they involve scores from multiple responses).

ABET outcome	Rating computed from EBI Alumni Survey Questions
A	(Q49+Q51)/2
B	(Q19+Q21+Q23)/3
C	Q25
D	Q27
E	(Q29+Q31)/2
F	Q33
G	(Q39+Q41)/2
H	Q35
I	Q47
J	Q8
K	Q37

Table I-5.9: Mapping between *a-k* outcomes and Alumni Survey Questions

A summary of responses received (percentage of yes and no responses) for some representative questions in the 2004 alumni survey is also shown below. These responses indicate that in the minds of our students, we are achieving reasonable success in imparting high quality, well-rounded education.

Question	**No**	**Yes**
Did your degree program meet its objective in giving the ability to apply your knowledge to the solution of practical & useful problems?	**4.76%**	**95.25%**
Did your degree program meet its objective in giving you the ability to communicate effectively and work collaboratively?	**4.76%**	**95.24%**
Did your degree program meet its objective in helping you become a successful professional in industry, government and, if you desire, in graduate studies?	**23.80%**	**76.20%**
Did your degree program meet its objective in helping you recognize the need for lifelong learning and the ability to adapt to rapid technological changes?	**15.00%**	**85.00%**
Did your degree program meet its objectives in helping you understand and deal with the ethical, societal, and global issues associated with the computing field?	**33.29%**	**66.71%**

Overall, the alumni survey scores were comparable to the scores obtained by the senior exit surveys. The alumni survey shows an improvement from 2002 to 2004 in categories *a-f* and *k*. In categories *h* and *i*, it shows a slight decline. However, due to the small number of respondents in 2002, the significance of this trend should be considered with caution. However, note that categories *h* and *i* are related (received broad education and life-long learning), and we have to consider ways to improve the scores in these two categories. Possible improvements are currently being examined and discussed.

While providing an opportunity to incorporate alumni of the Jonsson School into the outcomes assessment process has been a challenging task, our success in getting 44 responses in 2004 vs 8 responses in 2002 seems to suggest that we are becoming better organized in tracking our alumni. The Erik Jonsson School has recently hired an Associate Dean for Development who will help us in tracking our alumni's whereabouts better.

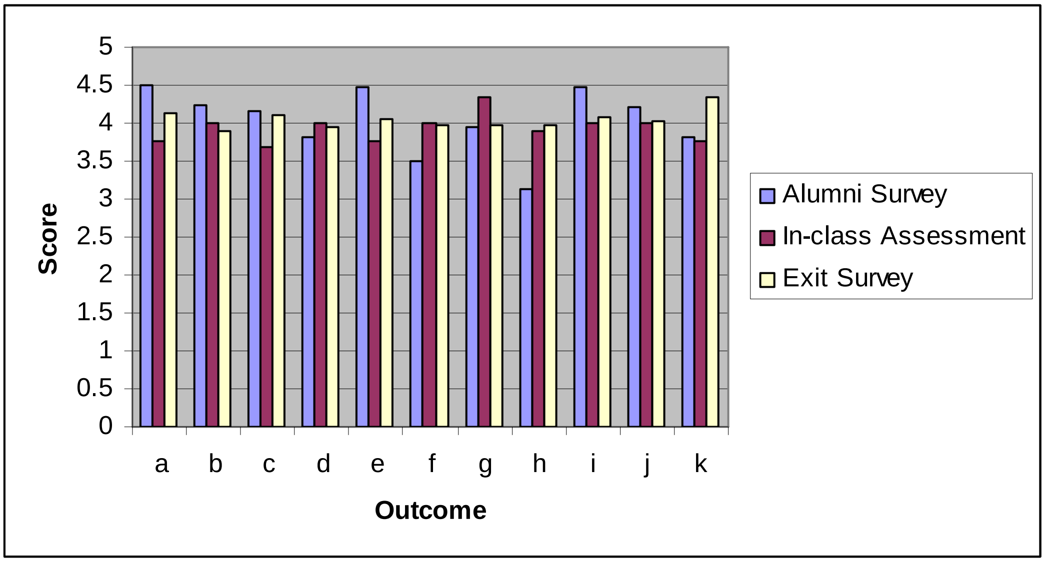


Figure I-5.9. Comparison of data collected for 2004 with three assessment devices. The exit and alumni survey scores are normalized to the range 1-5.

Given that the number of responses received from our alumni were low, in 2004 we decided to collect alumni feedback by assembling focus-groups of alumni. The first focus group meeting was conducted by Dr. Sook Kim, Assistant Dean for Assessment and Outcomes, on Oct 7th 2004. Note that the tenor of the discussion in the focus group was consistent with the written comments received from our alumni survey (that is, we heard similar praises and complaints). The focus group consisted of 8 alumni, who met with Dr. Kim for an hour and a half duration in two sessions. The group felt that the CS/SE program prepared them well for their jobs. In fact, they felt that they were more prepared than peers from other universities. There were a number of concerns, however. These concerns and the actions taken to rectify them are summarized below:

Different instructors teaching slightly different material in different offerings of the same course: To rectify this problem, a course coordinator has been appointed for each of the core classes to ensure uniformity across different sections of the course as well coordinate ABET activities. The course coordinators are overseen by the departmental curriculum committee and the ABET coordinator. Course instructors, through the Department Head, the curriculum committee as well as the course coordinators, have been instructed to keep the course materials for different sections as uniform as possible. However, note that two different professors teaching two different sections of the same course will never teach it in the same way, so these types of concerns are always going to be there, especially if one instructor is perceived as an "easy grader" compared to the other.

The degree program did not prepare adequately for technical writing: Our oral and written communications course, ECS 3390 is being strengthened. Also, all course instructors have been asked to introduce programming projects in their courses. This not only helps in improving the programming skills of the students but also help them improve their writing skills (through project report writing) and their teamwork skills (if team projects are involved).

The degree program is more theoretical and less applied: Several alumni felt that students are graduating without knowing how to program well. To rectify this all instructors have been requested to include programming projects in their classes. Additionally, the CS department has adopted a new grading policy wherein to pass a course that has programming assignments, a student has to pass both the theoretical component as well as the programming component. Earlier students could do well in the theory component, ignore the programming component and still could pass the class with a low grade.

Teachers engaging more with students in 1-to-1 interactions: The school has adopted a policy wherein every tenured/tenure track faculty member has to teach at least one undergraduate course every year. In addition, instructors have been asked to arrange problem solving sessions for each class, where either they or the teaching assistant can have closer interaction with the students. Action is on-going to designate Fridays as the day for recitation, when each class will hold its recitation session (the recitation session will not be obligatory so as not to increase the number of credits assigned to the course).

Employer Focus Group Meeting and Survey

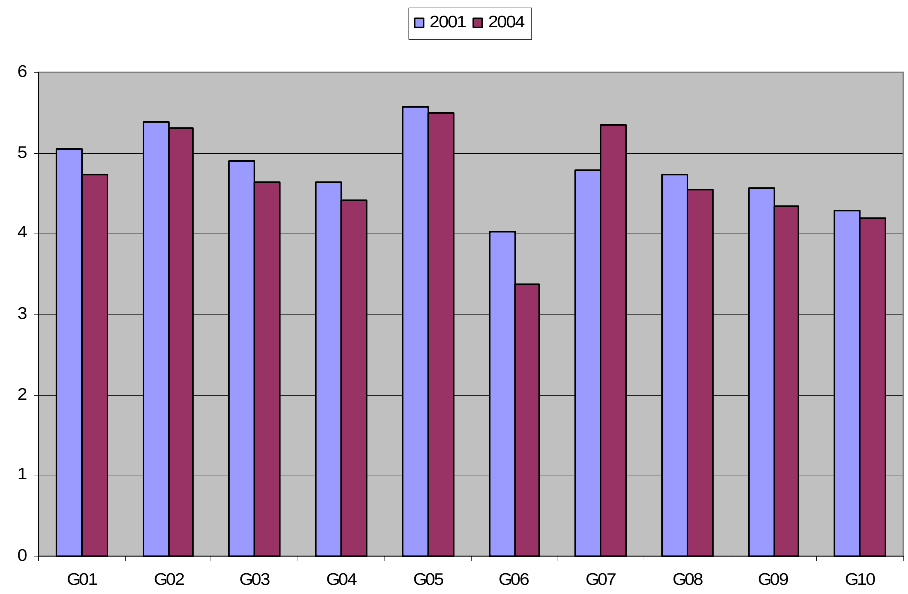
Continuous improvement of the curriculum remains an ongoing goal for colleges and universities. In particular, disciplines such as engineering and computer science demand not only the incorporation of current science and technological research, but also of new technological demand in industry. Moreover, one of the important missions of the Erik Jonsson School of Engineering and Computer Science is to meet the needs of our constituencies, including employers of our graduates.

The Office of Assessment first surveyed employers back in 2000 and then again in 2003-04. The next survey is currently under way with results expected in Fall 2005. The main focus of the employer surveys conducted in 2001 and 2004 was to evaluate changes in the Computer Science curriculum. Figure I-5.9 shows how the employer's rating of the performance of UTD graduates in different areas on a 1-6 scale.

Employer Focus Group (November 18, 2004): In the fall semester of 2004, the office of Assessment conducted an employer focus group to get feedback on the Software Engineering program. Participating employers (a total of 5) were selected from the set of supervisors of our BS graduates.

The feedback from the employers indicates that the program generally meets educational objective EO1. Graduates have broad knowledge of programming, algorithms, data structures but some concern was expressed in their ability to put it all together in a particular application.

The employers generally agreed that objective EO2 (communication) was met but pointed out that more can be done to help graduates understand and account for their audience. Note that this is consistent with what the alumni survey shows, that our graduates are less confident in their oral communication skills compared to their written communication skills.



G01	Communication Skills
G02	Critical Thinking
G03	Personal Management Skills
G04	Teamwork
G05	Ethics
G06	Contemporary Issues/Professionalism
G07	Problem Solving
G08	System Design and Implementation
G09	Data Analysis, Experimentation
G10	Probability and Statistics

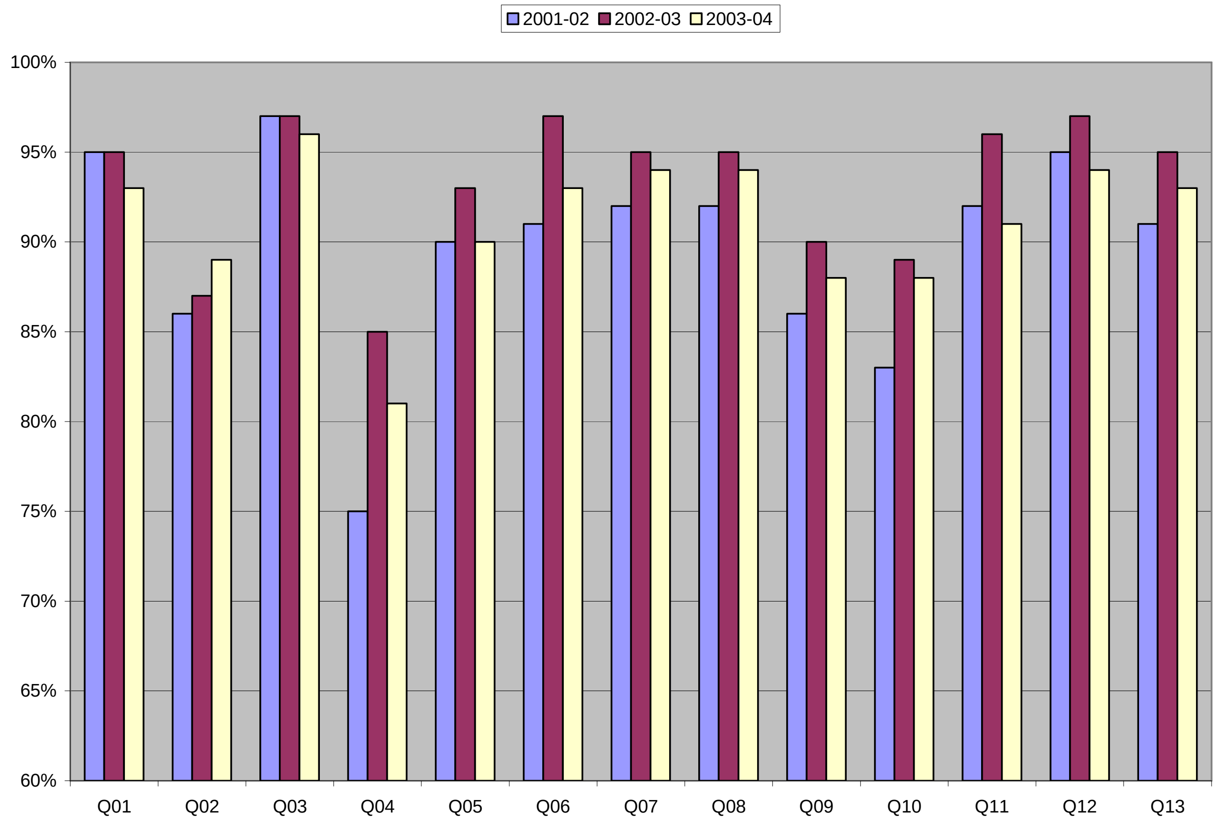
Figure I-5.9. Employer Surveys – Rating performance of graduates in specific areas 2001, 2004

Employers felt that our graduates did very well in meeting EO5 (social and ethical responsibility). The majority of employers felt the EO4 (need for life-long learning) was met but there were mixed opinions as some felt that their UTD graduate employees had not been in their jobs long enough for this criteria to be evaluated.

A concern was raised with respect to teaching of software design and architecture. The focus group findings were communicated to the relevant instructors in Spring 2005 and they are considering ways to improve their classes where these topics are covered.

IPP Satisfaction Surveys

Employers in the School's Industrial Practices Program were asked to participate in assessing students that were working in their companies. The satisfaction surveys were collected each year but covered both graduate and undergraduate students and did not distinguish among majors. The participating students cannot be viewed as true alumni as they have not graduated yet (some of the graduate students are probably alumni of UTD's undergraduate programs but they were not identified in these surveys). Neither can the survey be considered a true employer survey since expectations and duties of students participating in co-ops or internships may not be the same as those for full-time employees. Still, the data in Figure 5 does show that employers were generally satisfied with the quality of the students and do point out that an area of relative weakness is in communication skills.



Q01	Relations with Others
Q02	Judgment
Q03	Ability to Learn
Q04	Communication Skills
Q05	Technical Skills
Q06	Teamwork Skills
Q07	Dependability
Q08	Quality of Work
Q09	Quantity of Work
Q10	Educational Preparation for the Assignment
Q11	Potential for Greater Responsibility
Q12	Overall Performance
Q13	Performance Compared to Other Institutions

Figure I-5.10: Results of IPP Satisfaction Survey

Industrial Advisory Board (IAB)

In Fall 2003 we formed an Industrial Advisory Board for the Department of Computer Science and had its first meeting in November 2003. The discussions were very valuable and contributed to the introduction of new electives, modifications to the objectives and outcomes for the B.S. in Software Engineering, and the addition of ISSS 3360 (Social Issues) as a graduation requirement. The industrial advisory board also endorsed our program educational objectives.

The second meeting with the IAB was held in October of 2004. The meeting largely confirmed what our department was already practicing. The discussion in the meeting centered around outsourcing and off-shoring of software jobs and how that will affect the employability of our graduates. Suggestion was made that we should teach more system design in our classes and curriculum since these jobs cannot be outsourced. The IAB favored the approach of students "learning to learn" advising us not to focus too much on teaching specific tools. However, this is consistent with what the Department of Computer Science already practices. The emphasis on "design" and "learning to learn" was communicated to the faculty, and faculty advised to keep this in mind while designing their curriculum. However, it should be mentioned again, that this is something that the CS department already practices.

Standard I-6. The results of the program's assessments and the actions taken based on the results must be documented.

D. Program Improvement. Describe your use of the results of the program's assessments to identify program improvements and modifications to objectives. Include:

- any major program changes within the last five years
- any significant future program improvement plans based upon recent assessments

A large number of program improvements have been made as result of our assessment activities. Many of these improvements have been implemented, and some have even gone through another measurement cycle and showed tremendous improvement. These are summarized in the table below.

Date	Issue	Source	Resolution
04/02	Accreditation.	Feedback from industry, students, faculty	The faculty voted to seek ABET accreditation for both the BS-CS and BS-SE degree programs in 2005 (next visit for EE, TE)
2002-2003	Preparations for Accreditation		Set mission, objectives, outcomes. Adopted in-class assessment procedures from EE; started setting and measuring educational objectives.
04/03	Upgrade degree requirements	Feedback from ABET seminars; review of other BS-SE programs	Charge the Departmental Curriculum committee with studying the issue and making recommendations in Fall 2003.
2003-2004	Accreditation Processes	ABET requirements, faculty feedback, Industrial Advisory Board feedback	Established a Departmental Advisory Board which first met in Nov. 2003. Revised guidelines for in-class assessment several times. Worked with Office of Assessment to include CS, SE in its data collection/analysis activities.
20003-2004	Switch Programming classes from C/C++ to Java	Industry, student, faculty feedback	Gradual switch carried over 3 semesters is now almost complete
2004	Upgrade programming sequence	Student, faculty feedback	Added a Computer

Date	Issue	Source	Resolution
			Fundamentals class (not for CS,SE credit) to assure that students in the Computer Sci. I (CS 1337) class were at appropriate and uniform levels.
07/04	Clarify Science Elective; upgrade Probability and Statistics	Consulting visit for BS-CS	Tightened science elective requirement (list of acceptable classes); changed textbook and content of CS/SE 3341 to improve coverage of statistical concepts
07/04	Degree requirements in the catalog require less than 30 hours of broadening classes	Consulting visit for BS-CS	Improve breadth of curriculum by requiring 6 hours of electives in broadening areas; added to 2004-2006 catalog.
07/04	Readiness of Graduates for employment (software development skills)	Exit Surveys, student feedback	Selected classes were chosen to implement upgraded project requirements: Computer Science I (CS 1337) now starts with object-oriented design and added project work; Computer Science II (CS 2336) added a semester long project. Follow-up classes upgraded by including programming projects.
10/04	Graduates not proficient in programming; many avoid programming assignments and pass the course solely based on their exam grades.	Alumni survey, Student exit surveys.	Students have to pass both the theory as well as the programming portion of the class to get a passing grade.
02/05	Student populations in CS/SE 4340 (Comp. Arch.) have varied background due to different degree requirements making it difficult to cover the material in a uniform manner	In-class assessment; student feedback	Offer separate sections for CS4340 and SE 4340 in Fall 2005

Many further changes are planned in the future due to improvements and interventions contemplated recently. These include:

- Allowing only those with a C or better in the Programming Fundamentals class (CS 1336) to enroll in Computer Science 1 (CS 1337) class to ensure uniformity among majors and minors in CS (the department introduced a minor in Computer Science staring Fall 2005).
- Offering separate sections for Computer Architecture (CS 4340) for CS and SE majors (to be implemented from Fall 2005) due to the differences in the required classes that the CS and SE majors take (CS 2310, Digital Systems, is required for CS majors but not SE)
- Organize optional recitation sections every Fridays for all undergraduate classes.
- Organize activities such as Computer Fest (the best class projects are nominated for this competition) and programming competitions for learning outside the class and improved camaraderie among CS majors.

The specific changes to the program that have been implemented as a result of feedback received through surveys in last 4 years is summarized in the Table below (organized by categories). Note that in many instances we have been able to "close the loop," i.e., improvements identified have been made, and their positive impact measured.

Self-Assessment Feedback Cycle I	***Proposed interventions/ improvements***	***Results***	***Self-Assessment Feedback Cycle II***
Communication Skills			
2001 Employer Survey Curriculum needs more Communication skills related courses	***Fall 2001*** CS review and improve existing required writing courses by re-designing a course by A&H.	***Fall 2001*** Implemented the following: New Required course for all ECS majors: EE/CS 3390 Professional & Tech. Communication developed by A&H for ECS students.	**2004 SE/CS Employer Focus Group Meeting** Criterion (g) "Ability to communicate effectively" as a strength.
2002 Alumni Survey (1) "Preparation of communication" received the lowest factor mean (2) The highest difference in rating means between "important to career" and the "preparation" was "ability to communicate using oral progress tools"			**2004 Alumni Survey** (1) "Preparation of communication" improved from 2002 to 2004. (2) "Ability to communicate using oral & written progress reports rated low on performance ratings. (3) "higher importance" and the "lower performance" was found in "ability to communicate using written progress reports to your job or graduate school performance"
			2004 Exit Survey Ratings on communication skills improved from 2003 to 2004.

Self-Assessment Feedback Cycle I	***Proposed interventions/ improvements***	***Results***	***Self-Assessment Feedback Cycle II***

Career Advising			
2002 Alumni Survey "Degree Assisted in Career" factor mean was the second lowest out of 12 factors.	***Fall 2001*** Faculty advising procedure was modified. ***May 2002*** This was further improved with adequate Career and Degree advising by faculty advisors.	***Fall 2002*** Faculty advisors were given offices in the Office of Undergraduate Advising (OUGA) area to make OUGA "one stop shop" for advising.	**2004 Alumni Survey** "Degree Assisted in Career" was the most improved factor and received much higher mean than 2002. The difference in factors from 2002 to 2004 was the greatest among all 12 factors.
Fall 2000 Exit Survey Need more academic advisors.	***Fall 2000*** ECS undergraduate advising proposed organizational changes that include hiring more advisors. ***Spring 2002*** ECS undergraduate advising proposed organizational changes that include program coordinators.	***Summer 2001*** ECS undergraduate advising reorganized its structure and has 6 professional advising staff and the director ***Spring 2005*** ECS undergraduate advising reorganized its structure and has 7 professional advising staff and the Associate Dean.	**2002 Exit Survey** Significant improvement on the ratings of faculty advising from 2001 ratings **2003 Exit Survey** Significant improvement on the ratings of faculty advising from 2002 ratings **2004 Exit Survey** Decrease from 2003 ratings

Self-Assessment Feedback Cycle I	***Proposed interventions/ improvements***	***Results***	***Self-Assessment Feedback Cycle II***
Ethics			
2001 Employer Survey Curriculum needs exposure to work ethics, business ethics, and professional integrity.	***Spring 2001*** ECS and School of Social Sciences designed applied ethics within the context of engineering and technology.	***Fall 2001*** Implemented the following: New Required course for all ECS majors: ISSS-3360 Politics & Values in Business & Technology.	**2004 SE/CS Employer Focus Group Meeting** On a whole satisfied with the performance of our graduates in regards to ethical responsibilities.
2002 Alumni Survey "Preparation for ability to			**2004 Alumni Survey** Low performance rating was found in "ability to understand ethical

understand ethical responsibilities" was rated low.			responsibilities" **2004 SE/CS Alumni Focus Group Meeting** Not an adequate level. Need more context-based ethics course.
			2002 Exit Survey Improvement on the ratings from 2001 in "ability to understand ethical responsibilities." **2003 Exit Survey & 2004 Exit Survey** Although slightly lower ratings in 2003 in "ability to understand ethical responsibilities," but overall improvement on the ratings from 2002 and 2003 ratings.

Self-Assessment Feedback Cycle I	***Proposed interventions/ improvements***	***Results***	***Self-Assessment Feedback Cycle II***
Lab Facilities			
Fall 2000 Exit Survey Suggestions from our seniors: Increase the number of computers and/or computer labs.	***Fall 2002*** Build new labs and allocate substantial funds for buying computer equipment**.**	***Fall 2002*** Newly built ECS-South wing housed large computer labs: teaching labs with 130 workstations, two general purpose labs with 30 Sun Blade workstation, Graduate PC Labs with 30 stations, and Software Engineering Labs with 30 PCs	**2003 Exit Survey & 2004 Exit Survey** Satisfaction with quality of computing resources and use of laboratory improved steadily. In fact, the factor was the one which showed the most significant improvement in 2003 and 2004.

Self-Assessment Feedback Cycle I	***Proposed interventions/ improvements***	***Results***	***Self-Assessment Feedback Cycle II***
Academic Dishonesty			
Fall 2000 Exit Survey Suggestions from our seniors that we should prevent academic dishonesty. **2001 Exit Survey** Suggestions from our seniors that we should prevent academic dishonesty.	***2002-2004*** CS Department and Effective Teaching Committee discussed several methods. (a) Publish and disseminate our zero tolerance on Academic Dishonesty. (b) Scan student IDs and Photos prior to exams. (c) Mix several classes in a large classroom during the final exams. (d) Software to detect plagiarism. (e) Random seat assignment during finals.	***Fall 2003*** (a) Mix several classes in a large classroom during the final exams and/or move the classroom to larger venue. (b) Have TAs available for professors who request additional proctors. (c) CS: Implementation of plagiarism detection software (MOSS, etc.) (d) Publish our policy on academic dishonesty on the web (Fall 2002).	**2004 Exit Survey** Continue to receive suggestions from our seniors that we should prevent academic dishonesty.

Self-Assessment Feedback Cycle I	***Proposed interventions/ improvements***	***Results***	***Self-Assessment Feedback Cycle II***
Fellow Students			
2002 Exit Survey Level of camaraderie showed a drop over 2001. **2003 Exit Survey** All the questions Fellow Students showed a decrease in their mean values over 2002	***Fall 2002 & Fall 2003*** Propose to organize more activities centered around students	***Spring 2005*** Programming competitions organized; ACM Student Chapter made highly active (2002), ACE student research competition started (Spring 2004); Computerfest started (Spring 2005)	**2004 Exit Survey** The whole factor has improved over last year and the sub factor *Academic Quality* shows significant improvement.

Self-Assessment Feedback Cycle I	***Proposed interventions/ improvements***	***Results***	***Self-Assessment Feedback Cycle II***
Course Work			
2004 Exit Survey Suggestions from the seniors: Increase and provide more applied courses with practical skills.	***Fall 2003:*** Add more electives to the curriculum.	***Spring & Fall 2004:*** Several new courses added to the catalog and offered (real-time systems, computer security, etc).	***Feedback Expected in 2005 Exit survey***
Faculty			
Fall 2000 Exit Survey Suggestions from our seniors: Increase the number of faculty	***Fall 2001-4:*** Hire more faculty members to decrease the number of students per faculty**.**	***Spring 2002-5****:* Significant number of faculty candidates interviewed and hired.	T/TT faculty increased from 29 in Spring 2000 to 41 in Spring 2004

E. Program Evolution.

1. Describe in what respect, if at all, the philosophy and direction of computer science education has changed at your institution during the last five years (or since the last evaluation, whichever is the shorter duration).

The main change in philosophy, direction for the Computer Science Program over the past 5 years was to increase the design and software development experience in the curriculum while maintaining its focus on providing a solid foundation in Computer Science. The goal of the change was to better prepare students for successful careers after graduation. In terms of specific actions, we switched the main programming sequence from C++ to Java and introduced object-oriented design earlier, we added significant projects to several classes, and we introduced several Software Engineering classes that BS-CS students can use as guided electives. We also improved a number of advanced electives in diverse areas (Computer and Network Security, Image Processing, Real-time Systems). We also instituted a rule that to pass a course the student must pass both the theory and practice portions. Continuing along the same lines, the introduction of CS 1336 (Computer Fundamentals – no credit towards the BS-CS degree) serves to improve the level of the basic programming sequence by providing an outlet to students without the appropriate background (mostly non-majors). Also, the addition of several new electives in the 2004-06 catalog will further enrich the educational experience by providing hands-on experience with Embedded Systems, Computer and Network Security, implementation of Operating Systems and Networks.

Related changes focus on improved communication skills (by adding oral communication to ECS 3390 and increasing project report, presentation requirements in several classes), more exposure to hardware design (by replacing CS 2325 – Assembly Language with CS 2310 – Introduction to Digital Systems), and more coverage of ethical, social issues (by requiring ISSS 3360 and improving coverage in existing classes).

Other changes include requiring all tenured/tenure-track faculty to teach at least one undergraduate class every year. Normal teaching load for research active tenured/tenure-track faculty members is 3 courses per year. One of these must be an undergraduate class. The objective behind this action was to have research-active faculty members teach our undergraduate majors. Since UT Dallas has a large graduate program, many research-active faculty-members taught only graduate classes. Requiring all tenured/tenure-track faculty members to teach at least one undergraduate class every year will ensure that undergraduate students are exposed to research and new ideas as well.

The past year has been a very eventful one with the arrival of a new Dean and a major commitment and funding to upgrade the quality of the School. The drive for "top 50 in 5" is certain to entail significant changes in philosophy and direction but those are mostly in the formative stage at this time. One change in philosophy that is already taking hold is a switch in emphasis from growth (in the 1999 5-year plan) to quality improvement (in the "top 50 in 5" drive). While quality was certainly a component of the previous 5-year plan and growth is an important component of the current drive, it is fair to say that a significant shift in emphasis is taking place.

2. Describe any major developments and/or progress made in connection with the program in the last five years (or since the last evaluation, whichever is the shorter duration) that is not included in your response to I.C.

In 1999 the school undertook to writing a new 5 year plan. This plan complemented and expanded upon the then recently developed UTD Strategic Plan. In UTD's Strategic Plan, emphasis was placed upon service to the high-tech industry of North Texas and upon growth in the Engineering and Management Schools as the elements of UTD designated to serve the industry.

This strategic plan for the Erik Jonsson School of Engineering and Computer Sciencel guided the school in the period 1999-2004. The major objective of the School during the period was continued growth in enrollment combined with growth in funded research programs. Specific objectives included:

- Develop a program to improve the name recognition of the School of Engineering and Computer Science.
- Continue to focus research growth in the core technical competencies of North Texas: Telecommunications, Software Engineering, and Microelectronics; foster a research agenda that is complementary to the interests of local industry.
- Maintain the quality of the student body at all levels, while continuing to grow at existing rates, and improving the gender and the ethnic balance.
- Enhance all recruiting efforts to increase student quality, stressing community outreach programs and industrial partnerships; continue building a strong Co-op/Jobs program with local industry paying particular attention to the GSIIP.

Use modern information technology to enhance the School's programs at all levels, paying particular attention to the part-time graduate programs.

This 5-year plan has been proven very successful. For example, several new programs, which include BS, MS and PhD programs in Software Engineering, PhD program in Computer Engineering, and PhD program in Telecommunications Engineering, have been recently added to the School of Engineering and Computer Science. A joint BS program between EE and CS in the area of Computer Engineering is currently in planning stages. With respect to the undergraduate Computer Science program, there are several noticeable developments/progresses:

- Introduction of the fast-track program.
- New building, new research/teaching labs, and new classrooms.
- Major PC lab for students.
- Significant increase in student population and faculty size.
- Addition of CS 3354 Software Engineering to the core courses.
- Introduction of a number of new elective courses in diverse areas.
- Preparation for ABET accreditation.

F. Program Current Status.

List the strengths of the unit offering the computer science program.

The principal strength of the Department of Computer Science is the quality of its students and faculty. Most faculty members are professionally active publishing in leading research journals, serving on Editorial boards and conference Program Committees, and obtaining external funding. Student teaching evaluations are generally very good. Faculty size has more than doubled over the past 5 years. This, together with small increases in the number of senior lecturers, has led to a major reduction of classes (to nearly zero) taught by part-time lecturers in accordance with our goal to improve consistency and quality in undergraduate teaching. Recent departmental rules requiring all faculty members to teach undergraduate classes has resulted in a richer educational experience for our undergraduates.

The Department directly benefits from the outstanding personal and financial support the Engineering and Computer Science School receives from the neighboring high-technology community. The Dallas-Fort Worth community raised over $24,000,000 in private funds to support the initiation of the School. Similarly, $40,000,000 was recently raised to support the construction of a new wing (occupied mostly by the Department of Computer Science since its completion in Fall 2002). A five year, $300 million initiative, called the Jonsson School Research Excellence Initiative, supported by the State of Texas, the UT-System, Texas Instruments, and other entities was announced in June 2003 with the goal of enhancing Engineering and Science education and research at UT-Dallas. Included in this initiative are a new Laboratory facility (expected in 2006), and funding for 40 chaired professor positions and 400 PhD students. Local industries including Texas Instruments, Alcatel, Ericsson, Nortel, Fujitsu, Nokia, and Raytheon support significant research programs in the School, and significant gifts of equipment have come from local donors.

Another strength of our Department is the curriculum, comprising a strong mathematical foundation and core computer science studies. In addition, students choose their elective courses from a wide variety of courses ranging from numerical analysis, database systems, computer graphics, artificial intelligence, object-oriented programming systems, compiler design, to computer networks. This is possible because of a large and diverse (and still expanding) faculty. Furthermore, students may also choose electives from Electrical Engineering such as VLSI and microprocessor design. New electives were developed over the past year and included in the 2004-06 catalog. They include courses in Computer Vision, Computer Animation, Computer and Network Security, Embedded Computer Systems, Implementation of Modern Operating Systems, and a Networking Laboratory. These electives give students a strong preparation for continuing graduate education or for direct entry into industry.

Many of our students, both at the graduate and undergraduate level, are employed in local industry. These students come to their studies with a seriousness of purpose and a depth of experience that enriches the learning environment. This mature and motivated student body is another strength of our Department. Our undergraduates have developed a strong "esprit de corps" and use the ACM and IEEE student chapters as an effective mechanism for promoting student activities and student-faculty social interaction. Recently added events such as Computer Fest (where the best projects from undergraduate classes, nominated by instructors, compete with each other) have further strengthened this camaraderie. With entering SAT scores among the highest in the State and a renewed emphasis in quality undergraduate education, the School of Engineering and Computer Science is well positioned to become nationally recognized.

A full co-op program (one of the 5 largest in the nation) actively places students into meaningful jobs in local industry. This program is another strength that offers our students both excellent training and permanent careers. Our graduates to date have been well placed in local industries.

The Undergraduate Advising office has been very effective in providing a central location where all undergraduates in the School of Engineering and Computer Science can obtain assistance and guidance.

The honors CS program, which offers an alternate set of courses for qualified undergraduates, has been very well received and plans are underway to increase the number of honors sections offered.

List any weaknesses or limitations of the institution or unit offering the computer science program.

The limitations in our Department fall into two main categories. The first set of limitations is related to our rapid growth as a Department, and the second is related to our program itself.

Past and projected enrollment data shows that enrollment in the Department has grown very rapidly at a 16.5% rate per year since 1995. The funding mechanisms in the UT System are not suited for rapid growth, and therefore, the resources available to the Department have been lagging behind. Furthermore, the recent economic downturn has placed additional strain on budgets. We have identified the following weaknesses in our program:

1. While we have a Senior Design course in our curriculum, it is currently not required in our undergraduate program. Design and software development experience are identified in exit surveys and in the Industrial Advisory Board meetings as important factors that need to improve. The popularity of the Software Engineering Program (and double majors in Computer Science and Software Engineering) is a clear indication that students want more design and development experience in the curriculum. The curriculum committee is looking into adjustments to the degree requirements that will integrate a mandatory senior design project into our program (plan to be presented to the faculty in Fall 2005).

2. Feedback from students, alumni, and employers points to a continuing perception that the Computer Science Program is "too theoretical". There is a risk that this perception will be strengthened by the publicity surrounding the "top 50 in 5" initiative (which is likely to emphasize research, external funding, PhD programs). While we have taken corrective action (as outlined above) and the "top 50 in 5" initiative actually includes a strong commitment to high-quality undergraduate education (e.g., more tenure-track faculty teaching undergraduate classes, plans to improve retention of our students for graduate study, recognition that quality improvements at the graduate and undergraduate levels are codependent), it may well be that concrete steps need to be taken to correct this perception.

3. The Professional and Technical Communications class (CS 3390) has been very useful. But more emphasis needs to be placed on improving oral communication skills beyond the experience gained in Rhet 1302 and this class. This improvement can be done by requiring formal presentations in some CS classes (e.g., project presentations). While some of this emphasis on class projects has been incorporated in the program recently, we still have more work to do, given that the 2004 alumni survey identified oral communication as still a major weakness. The possible remedies are being debated and we hope to make some concrete proposals to our Faculty in the Fall 2005 semester.

4. While doubling faculty size within the last 5 years, the student/faculty ratio is still large. Enrollments at the graduate level have leveled off and the pace of growth in the Undergraduate Program has slowed down, however, our faculty to student ratio still remains high. This, together with the addition of new faculty (4 new hires starting from Fall 2005 alone), and plans to have more tenure-track faculty teach at the undergraduate level should improve student/faculty ratios.

List any significant plans for future development of the program.

Plans stemming from the "top 50 in 5" drive are in the formative stage. Some that are already in place or will likely be implemented soon include enriching the undergraduate educational experience by having more tenure-track faculty teach undergraduate classes, involving interested undergraduates in research early in their studies and making a concerted effort to retain highly qualified students for graduate study (Ph.D.) through our GetDoc program.

In Spring 2005, we organized a new event for Undergraduate students called the "Computerfest," in which the best class projects compete for awards. Nominated projects are presented by their student designers in front of an audience consisting of students and faculty. The first Computerfest was very well received. We plan to make this a regular major event,every semester. The Computerfest will help the students in broadening their knowledge as well as in improving their presentation skills.

II. Student Support

INTENT: Students can complete the program in a reasonable amount of time. Students have ample opportunity to interact with their instructors. Students are offered timely guidance and advice about the program's requirements and their career alternatives. Students who graduate the program meet all program requirements.

The ***Intent*** must be met in order for a program to be deemed accreditable. One way to meet the ***Intent*** of this criterion is to satisfy each one of the **Standards** listed below. To do this, answer the questions associated with the **Standards**. If one or more **Standards** are not satisfied, it is incumbent upon the institution to demonstrate and document clearly and unequivocally how the ***Intent*** is met in some alternative fashion.

If you are having more than one program evaluated, particularly if the programs are on this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

Standard II-1. Courses must be offered with sufficient frequency for students to complete the program in a timely manner.

A. Frequency of Course Offerings.

1. List below the course numbers, titles, and credit hours of courses required for the major which are offered less frequently than once per year. Explain how it is determined when they will be offered, e. g., rotation, odd-numbered years, or whatever.

All of the required courses are offered each Fall and Spring Semester. Frequently, multiple sections of these core courses are offered in each semester. In addition, many required courses are offered during each Summer Semester. Effort is also made to offer at least one section of each each class in the evening (after 4pm) at least once every year (we are able to meet this objective for 99% of our classes). If a core class has multiple sections, at least one section is offered in the evening.

2. List below the course numbers, titles, and credit hours of courses allowed for the major but not required (i. e., either free electives or lists of courses from which students must choose a certain number), and explain how it is determined when they will be offered.

The elective courses are offered according to demand and availability of faculty to teach them. For popular electives such as Databases (CS 4347) and Object Oriented Programming Systems (CS 4376) multiple sections were offered in the Fall and Spring semesters.

Course	Number of times offered in, Spring 2004, Summer 2004 and Fall 2004 semesters
CS 4334 Numerical Analysis	1
CS 4336 Advanced Java Programming	0
CS 4347 Database Systems	3
CS 4361 Computer Graphics	1
CS 4365 Artificial Intelligence	1
CS 4376 Object-Oriented Programming Systems	3
CS 4380 Senior Design Project	2
CS 4386 Compiler Design	0
CS 4390 Computer Networks	3
SE 4351 Requirements Engineering	2
SE 4352 Software Architecture and Design	2
SE 4367 Software Testing, Verification, Validation and Quality Assurance	3
SE 4485 Software Engineering Project	3

Table II-1.1: Computer Science elective courses.

Standard II-2. Computer science courses must be structured to ensure effective interaction between faculty/teaching assistants and students in lower division courses and between faculty and students in upper division courses.

B. Interaction with Faculty.

1. Describe how you achieve effective interaction between students and faculty or teaching assistants in lower division courses, particularly in large sections.

The department does not use teaching assistants in the usual sense. All courses are taught by faculty. Use is made of teaching assistants as graders and to answer questions, and some may provide additional assistance in preparing materials. Teaching assistants are required to hold office hours. They are assigned offices for holding these offices hours.

Faculty are required to have sufficient hours of availability in their offices, though most usually are available numerous hours beyond that. The office hours of each faculty member are posted outside his/her office at the beginning of each semester by the departmental staff. Also, a number of the faculty schedule extra review sessions and tutorials. For many courses, such as Computer Science I and II, faculty members and TAs also schedule hours in the Lab where students can come and seek help for their homework and/or projects.

Course-related web sites provide a convenient mechanism for interaction outside classes. Several faculty members have created web sites varying in sophistication from merely providing office hours and syllabi to comprehensive sites offering a full range of student support materials. The University supports instruction through WebCT which provides mechanisms for the instructor to communicate with all or a subset of the students, as well as for students to discuss things among themselves. The Information Resources (IR) division of the University automatically sets up the WebCT for each course offered at the beginning of each semester. The list of students and instructor names are obtained from the Registrar's office and automatically entered into WebCT by the IR division.

2. Describe how you achieve effective interaction between students and faculty in upper division courses. Give detailed explanation and/or documentation how you do this for sections with more than thirty students, if applicable.

The comments above apply to lower and upper division courses.

Standard II-3. Guidance on how to complete the program must be available to all students.

C. Student Guidance. Describe what determines the requirements that a student will follow and how the student is informed of these requirements.

The curriculum is designed so that all ABET requirements are fulfilled solely by taking the required courses. The requirements for obtaining a BS in Computer Science are laid out in our undergraduate catalog which is updated on the Web every year (hard copy is published every 2 years). The Undergraduate catalog is widely disseminated through the Web (Appendix VIII contains the CS section of the Undergraduate catalog). Additionally, the undergraduate academic advisors as well as the faculty advisors and the Associate Dean for Undergraduate Education (ADUE) review the student record each semester to ensure compliance and satisfaction of requirements. For graduating seniors, each record is carefully reviewed by the ADUE and the Registrar's office to ensure that all curriculum requirements are met. Students may petition to substitute courses taken elsewhere for required courses, but permission is granted only if the content of the course considered is essentially the same as the course to be replaced as verified by catalog descriptions, course syllabi and textbooks used.

The University has a staff of Academic Advisors (called the Office of Undergraduate Advisors or OUGA) to assist students in designing an appropriate course of study that will satisfy requirements for graduation. Since the faculty is most knowledgeable about course content and course decisions that lead to specific career areas, the Computer Science Department has appointed additional advisors who are chosen from faculty ranks and who also advise each student. These faculty advisors are Computer Science faculty members who are given a reduced teaching load in lieu of their advising duties. The faculty advisors hold regular office hour in the OUGA office suites where they advise students. The office hours of faculty advisors are widely available. Office hours are set to coincide with peak student traffic periods.

All degree-seeking students are encouraged to meet regularly with their Academic Advisor (both faculty and professional advisors) to discuss:

- Class Selection
- Difficulties with Course Work
- Degree Requirements
- University Rules and Procedures
- Time Management
- Study Skills
- Career Opportunities
- Personal Problems

The academic advisors also fill out the degree plan that is placed in a student's file, to which all advisors have access. The degree plan is updated from time to time as the students' plans change. The updates are made after the student has met an advisor and discussed the changes. A blank degree plan is included in Appendix IX:

Faculty Advisors for Computer Science

Lawrence A. King	6612	lak022000@utdallas.edu	ECSS 3.701
Marth Sanchez	4723	mxs015000@utdallas.edu	ECSS 3.703
Raphael Lacambra	4824	rml021000@utdallas.edu	ECSS 3.704
Joseph Leubitz,	2854	jleubitz@utdallas.edu	ECSS 3.705
Laurie Thompson,	4839	lthomp@utdallas.edu	ECSS 3.610

Academic Advisors for Computer Science and Electrical Engineering

Sandy Bowen, MS	6846	sbowen@utdallas.edu	ECSS 2.502
Cathy Hill, MS	6848	cathyh@utdallas.edu	ECSS 2.502
Carol Nguyen, MS	4803	carolhn@utdallas.edu	ECSS 2.502
Mary Ann Stewart,	2108	Mas051000@utdallas.edu	ECSS 2.502
David Kemerling, MS	2359	David.Kemerling@utdallas.edu	ECSS 2.502
April Liang, MS	6224	aliang@utdallas.edu	ECSS 2.502
Kenneth Richards, MS	4181	kenrich@utdallas.edu	ECSS 2.502

For each advisor, the phone extension and email is shown along with their office location (OUGA is located in Suite 2.502 of the Engineering and Computer Science Complex South building which houses the computer science department Each advisor has an office in this suite).

OUGA has a comprehensive website that is also widely disseminated through the WEB (http://www.utdallas.edu/dept/eecs/ouga/index.html).

Standard II-4. Students must have access to qualified advising when they need to make course decisions and career choices.

D. Student Advisement. Describe your system of advisement for students on how to complete the program. Indicate how you ensure that such advisement is available to all students.

The Office of Undergraduate Advising (OUGA) in the School of Engineering and Computer Science has a staff of seven professional Academic Advisors and five faculty advisors to assist students as they progress toward completion of requirements for graduation. The academic and faculty advisors are the primary contact for students in developing academic goals, creating and maintaining degree plans, developing final degree plans for graduation, obtaining referrals to faculty and university offices, and answering questions or working to resolve academic or personal issues. The advisors are very knowledgeable about the university at large as well as changes and information relevant to specific curriculums. Any policy or curricular change is quickly disseminated to the advisors through the Curriculum Committee. The curriculum committee is structured so as to include at least one faculty advisor.

Each student is assigned to a specific academic advisor (the caseload is divided based on the first letter of the last name). Prior to Spring 2005, a student could see any academic advisor, however, many student fell through the cracks and never saw any advisor if they so chose. Academic advisor are present in the OUGA office from 8AM to 5PM on all working days. With respect to faculty advisor, a student can see any one of them depending on their availability. Faculty advisors hold their advising office hours in the OUGA offices; these office hours are widely publicized. Advisors are also available for routine information through email and phone. An advising file is set up for each student so that all advisors have access to complete information for the student to expedite the advising process. In addition, a contact log is maintained. Each contact made with a student, regardless of the method, is entered in this log with brief notes regarding the contact context. Advisors across the University have access to this contact log in order to facilitate sharing of information as students are served.

The faculty advisors as well as the professional advisors are prepared to address a broad range of issues that a student may present. Each advisor is knowledgeable about degree requirements for all degrees offered. In addition, each advisor is familiar with university policies that could impact students' academic progress. The advisor will discuss potential problems with the student and facilitate resolution to these problems through referrals to appropriate university offices or consultation with appropriate university staff to correct the problem.

A degree audit is available to all students upon request. The advisor can discuss course options that will satisfactorily fulfill specific requirements as well as pre-requisite and co-requisite requirements. The student is encouraged to discuss course selections that will support specific career goals with faculty. The advisor assigns transfer credits to appropriate degree requirements for new students. If the assignment is not apparent, the advisor accepts course documentation to support petitions submitted to the Associate Dean for Undergraduate Education for consideration. The advisor is then responsible for entering approved transfer substitutions into the student's record so that the degree audit accurately reflects the application of all transfer credits. In addition, each advisor facilitates course enrollment during registration periods, including resolution of issues resulting from transfer credit application, e.g., overriding a pre-requisite when a transfer credit has been used that does not reflect the specific UTD course number.

Degree audits are done for all graduating students before submitting their file to Records for the final certification audit. Advisors review the degree requirements when the student submits an application for graduation to ensure that all requirements will be met with the proposed enrollment for that semester. If this review is satisfactory the advisor signs the application for graduation so the student can officially apply for graduation. Before Census Day the applicants' files are again reviewed for degree requirement completion. If problems are identified the student is contacted and efforts are made to resolve the problem to ensure the student's graduation. After this second review the list of graduation applicants is submitted to The Office of Records for the final audit leading to official graduation certification.

At all times throughout the year the faculty and professional advisors are available to students who present personal or academic issues that are interfering with academic performance. The advisor will talk with the student to determine if the advisor can adequately assist in resolving the issue, or if the problem might best be resolved utilizing another university resource. If another resource seems more appropriate the advisor will make the referral and make sure the student has sufficient information to successfully follow through. If the advisor feels the problem can be resolved using the advisor's expertise and the established relationship between the student and the advisor, the advisor will develop a time frame and approach, in conjunction with the student, to seek a solution. In specific situations the advisor will be expected to schedule regular appointments with a specific student group to discuss a commonly shared problem, e.g., scholarship students who are on probation for failing to meet the minimum scholarship requirements.

OUGA offers a more formally structured opportunity called Operation Improved Strategies for students whose academic performance has been deemed inadequate by the University. This program is designed to provide group support for the participating students, and offers opportunities to explore the personal responsibility each student must assume for their academic standing, areas needed to change, and personal and university resources available to assist in these changes. Participation in this program will be a factor considered by the Associate Dean for Undergraduate-education (ADU) when the student requests permission to enroll in the next semester as required by the University.

Since faculty is most knowledgeable about course content and course decisions that lead to specific career areas, it is important that students have access to them quickly and easily. In order to supplement and facilitate access to faculty advisors, OUGA has provided offices for faculty advisors within the office suite for Student Services. The schedule for faculty advisors is posted in OUGA and provides availability during the peak student traffic periods. Using this design the students have access to a faculty advisor quickly in a location they are already familiar with when they have questions regarding career decisions or curriculum choices.

The faculty advisors concentrate more on assisting students with course selection, study plans, and career planning. Their knowledge of the curriculum, and of specific course contents enables them to answer questions relating to a student's readiness to take a class, material that a student should brush up on, or what the student should concentrate on in a particular class and with their particular background. They can recommend specific electives that fit the student's background and career plans. They bring to the task a wealth of industrial experience and can assist students with questions about the job market, and career planning.

The faculty advisors work closely with the academic advisors, however, the two groups are largely complementary in the services they provide to the students with domain expertise being the critical difference between the two groups. Also, it should be noted that all faculty are informally involved in advising since students may approach their instructors with some of the same questions that they would go to the designated faculty advisors with; also, it is fairly common that those formally charged with advising, will often interact with the faculty to resolve course specific issues that might come up.

Reference: OUGA http://www.utdallas.edu/dept/eecs/ouga/index.html

Learning Resource Center

The Learning Resource Center offers assistance to students in the areas of reading, writing, mathematics, and study skills through individual appointments, group workshops, short courses, and audio and video tapes. The Writing Lab offers one-to-one assistance with writing assignments and general writing skills. The Math Lab gives short-term and semester-long support for a variety of mathematics courses. The Learning Resource Center also offers developmental math, reading, and writing classes. These classes are for credit, but they do not count toward graduation. Assistance is also available in study skills, note-taking, writing, test taking, algebra, and preparation for the TASP (required for teacher certification), GRE, GMAT, and LSAT. In addition, students can receive help with time management, basic mathematics improvement, test-anxiety reduction, and various other study techniques and strategies. All students enrolled at the university are eligible for these services.

E. Access to Qualified Advising. When students need to make course decisions and career choices, what is their procedure for obtaining advising? Do they have adequate access to qualified professionals when necessary?

(see C and D above)

Standard II-5. There must be established standards and procedures to ensure that graduates meet the requirements of the program.

F. Meeting the Requirements. Describe your standards and procedures for ensuring that graduates have met all of the requirements of the program.

(see C and D above)

III. Faculty

INTENT: Faculty members are current and active in the discipline and have the necessary technical breadth and depth to support a modern computer science program. There are enough faculty members to provide continuity and stability, to cover the curriculum reasonably, and to allow an appropriate mix of teaching and scholarly activity.

The ***Intent*** must be met in order for a program to be deemed accreditable. One way to meet the ***Intent*** of this criterion is to satisfy each one of the **Standards** listed below. To do this, answer the questions associated with the **Standards**. If one or more **Standards** are not satisfied, it is incumbent upon the institution to demonstrate and document clearly and unequivocally how the ***Intent*** is met in some alternative fashion.

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

If different programs have different faculty members, please identify which faculty are associated with which program(s), and the percentage of time allotted, if they are associated with more than one.

Standard III-1. There must be enough full-time faculty members with primary commitment to the program to provide continuity and stability.

A. Faculty Size. The purpose of this section is to determine whether you have sufficient faculty to offer courses often enough for students to complete the program in a timely manner.

In the previous section you gave the course numbers of courses required for the major which are offered less frequently than once per year, and those allowed for the major but not required, and explained how it is determined when they will be offered. Explain (if applicable) any difficulties you have offering required or optional courses frequently enough, particularly as they might be affected by faculty size.

Since 1998, the Computer Science faculty size has almost doubled. The growth in faculty size has occurred faster than the growth in the number of students. We do not have the problem of not offering any required or elective courses frequently enough.

B. Faculty with Primary Commitment.

1. Read the definition of "Primary Commitment" in the *Guidance* (Section III, point 3) and list here the number of faculty whose primary commitment is to this program: _20____.

The purpose of the next questions is to ascertain the degree of continuity and stability provided by these faculty.

2. Please list below the number of faculty with primary commitment to the program in each academic rank, broken down within rank by tenure status.
3.

The following is the list of permanent faculty members of the Department. All the names marked with * taught undergraduate courses in the last 2 years. Senior lecturers are primarily committed to the undergraduate program. However, tenured and tenure-track faculty members are actively involved, and some play leadership roles in undergraduate teaching.

Professors, Tenured

Dr. Farokh Bastani*, Excellence in Education Chaired Professor
Dr. R. Chandrasekaran*, Ashbel Smith Chaired Professor.
Dr. Ding Z. Du
Dr. Andras Farago*
Dr. Gopal Gupta, Associate Department. Head
Dr. D.T Huynh*, Department Head
Dr. Dan Moldovan
Dr. Simeon Ntafos, Associate Dean for Undergraduate Education
Dr. Balaji Raghavachari*, Assistant Department Head
Dr. Edwin Sha*
Dr. I. Hal Sudborough*, Founders Chaired Professor.
Dr. Bhavani Thuraisingham
Dr. Klaus Truemper*
Dr. Si Qing Zheng*

Associate Professors, Tenured

Dr. Sergey Bereg (tenure-track)*
Dr. Lawrence Chung*
Dr. G.R Dattatreya*
Dr. V. Hatzivassiloglou
Dr. Sanda Harabagiu
Dr. Ivor Page*
Dr. Ravi Prakash*
Dr. Rym Mili*
Dr. Haim Schweitzer*
Dr. Subbarayan Venkatesan*
Dr. Eric Wong*
Dr. I-Ling Yen*
Dr. Yuke Wang*
Dr. Kang Zhang*

Assistant Professors, Tenure Track

Dr. Joao Cangussu*

Dr. Jorge Cobb*
Dr. Kendra Cooper*
Dr. Ovidiu Daescu*
Dr. Jing Dong*
Dr. Jason Jue*
Dr. Latifur Khan*
Dr. Neeraj Mittal*
Dr. Vincent Ng*
Dr. Balakrishnan Prabhakaran*
Dr. Kamil Sarac*
Dr. R.N. Uma*
Dr. Weili (Lily) Wu*
Dr. Youtao Zhang*

Senior Lecturers

Tim Farage*	(MS)
Dr. Herman Harrison*	(PhD)
Sam Karrah*	(MS)
Lawrence King*	(MS)
Dr. Radha Krishnan*	(PhD)
Rafael Lacambra*	(MS)
Joseph Leubitz*	(MS)
Greg Ozbirn*	(MS)
David Russo*	(MS)
Martha Sanchez*	(MS)
Cort Steinhorst*	(MS)
Dr. Anthony Sullivan*	(PhD)
Laurie Thompson*	(MS)
Nancy Van Ness*	(MS)
Dr. Wei Wei*	(PhD)

Standard III-2. Full-time faculty members must oversee all course work.

Standard III-3. Full-time faculty members must cover most of the total classroom instruction.

C. Faculty Oversight. Full-time faculty must oversee all course work allowed towards the major. That means that they must either teach a course or be the course chairperson or coordinator for all sections taught by other than full-time faculty, such as adjunct faculty or teaching assistants. For those courses with sections not taught by full-time faculty during the past academic year, list the course numbers below and the name of the full-time faculty coordinator. (The past academic year is the academic year immediately prior to the year in which this report is prepared.)

The number of courses taught by adjunct faculty is nearly zero. All faculty members listed above are full-time and teach all our classes. A set of faculty members is assigned to each core course, responsible for setting the learning objectives and the assessment of the course (Table III-3.1; course coordinator is shown in bold)

Course	Faculty
CS1337	**Steinhorst**, Farage, Harrison, Page, Sanchez
CS2305	Van Ness, Cooper, **Farage**
CS2310	**Harrison**, Page, Zhang
CS2336	Harrison, **King**, Sanchez, Steinhorst
CS3305	Farage, Krishnan, **Van Ness**, Zheng
CS3345	Daescu, Ozbirn, Page, Prakash, Sanchez, **Uma, Bereg**
CS3354	Leubitz, Sullivan, **Chung**
CS4340	Thompson, **Dattatreya**
CS4348	Venkatesan Yen, **Ozbirn**
CS4349	**Daescu,** Sudborough, Wei, Krishnan
CS4384	**Huynh**, Sudborough, Van Ness
CS4390	**Cobb,** Sarac, Farage

Table III-3.1: Faculty members assigned to courses

Standard III-4. The interests and qualifications of the faculty members must be sufficient to teach the courses and to plan and modify the courses and curriculum.

Standard III-5. All faculty members must remain current in the discipline.

Standard III-6. All faculty members must have a level of competence that would normally be obtained through graduate work in computer science.

Standard III-7. Some full-time faculty members must have a Ph.D. in computer science.

D. Interests, Qualifications, Scholarly Contributions. The criteria state that the interests, qualifications, and scholarly contributions of the faculty must be sufficient to teach the courses, plan and modify the courses and curriculum, and to remain abreast of current developments in computer science. This information should be contained in the faculty vitas attached to this report and need not be repeated here. This would be an appropriate place to insert a description of general departmental or institutional activities that promote faculty currency, if such exist. (A sample vita questionnaire is attached in section G below. Although it is not necessary to follow this format, it is important that whatever format is followed contain all the information asked for. And, to make things easier for the visiting team, please see that all faculty vitas are in the same format, whichever format is used.)

UTD emphasizes on both teaching and research. Research is critical to the Department. The departmental mission statement says:

> "…. The Department places high priority on establishing and maintaining innovative research programs to enhance its education quality and make it an important regional, national and international resource center for discovering, integrating and applying new knowledge and technologies."

Active research maintains faculty currency. 75% of our faculty is tenure-track and is expected to maintain an active research program. 50% of such a tenure-track faculty member's time is expected to be devoted to research and scholarship, which helps in keeping these faculty members current. Non-tenure track faculty members stay current by attending departmental seminars, our school's distinguished lecture series. A few even engage in research projects. Starting from Spring 2005, they are required to attend at least one technical conference a year, which is paid for by the Department/School.

Standard III-8. All full-time faculty members must have sufficient time for scholarly activities and professional development.

E. Scholarly Activities. Describe the means for ensuring that all full-time faculty members have sufficient time for scholarly activities and professional development.

On average, tenure track faculty allocate 40% of their effort to teaching, 50% to research and 10% to other activities (service). Faculty with administrative duties allocates 20% to teaching, 40% to research and 40% to administration/service. Senior lecturers usually allocate 80-100% on teaching. Many of the senior lecturers engage in research on their own accord (principally, in the summer), even though research is not required of them.

The Department has a very active weekly seminar series. Researchers and industrial leaders are invited to present new results and ideas. Additionally, the school has a distinguished lecture series, in which world renowned researchers (Turing award winners, National Academy of Engineering members, Fellows of the ACM and IEEE) are invited to lecture. Tenure-track and non tenure-track alike attend these lecture series.

The faculty, including senior lecturers, who are conducting research projects with external funding are given reduced teaching load. Faculty members with funded projects can also "buy out" teaching of one course per year. UT Dallas also supports the Special Faculty Development Assignment (SFDA) program (which is similar to a sabbatical program) in which faculty members can take a 6 months to a year leave of absence to focus on research related activities.

In order to help the professional development of non-tenure track faculty members, the School requires them to attend at least one technical conference each year (the expenses are paid by the Erik Jonsson School).

Standard III-9. Advising duties must be a recognized part of faculty members' workloads.

F. Support for Advising. Advising duties must be a recognized part of faculty members' workloads, which means that faculty with large numbers of advisees must be granted released time. Explain your advising system and how the time for these duties is credited.

Faculty members who are assigned as undergraduate advisors have reduced teaching load of one course per semester.

G. Information Regarding Faculty Members.

On separate pages, please furnish the following information for all faculty members that teach courses allowed for the major, including those who have administrative positions in the department (Head, Associate Head, etc.). Use the form given below as guidance. This form need not be followed exactly, but all the information asked for should be supplied. Please do use a common format for all vitas. Please limit information to no more than three pages per person, if at all possible. Please place the form(s) for administrator(s) first, followed by the others in alphabetical order.

In case more than one program is involved, especially with separate campuses, please indicate clearly the program(s) an individual is assigned to, and the percentage of time to each, if more than one.

Within the Department, there are two closely related programs, Computer Science and Software Engineering. A faculty member contributes to both programs. Summary CVs of all faculty members are included in Appendix X. However, a synopsis of our faculty is given in Table III-9.1 below:

Table III-9.1: Faculty Analysis

Name	Rank	FT Or PT	Highest Degree	Institution from which Highest Degree Earned & Yr.	Years of Experience			Level of Activity (high,med,low,none) in:		
					Govt./ Ind. Practice	Total Faculty	At UTD	Prof. Society	Research	Consult/ Summer Work in industry
F. Bastani	Professor	FT	Ph.D.	Berkeley '80	0	25	8	High IEEE, ACM	High	None
S. Bereg	Associate Professor	FT	Ph.D.	Minsk '92	4	8	3	High ACM	High	None
J. Cangussu	Assistant Professor	FT	Ph.D.	Purdue '02	0	7	3	High ACM, IEEE	High	None
R. Chandrasekaran	Professor	FT	Ph.D.	Berkeley '67	2	36	30	High SIAM IEEE, ACM	High	Medium
L. Chung	Associate Professor	FT	Ph.D.	Toronto '93	2	12	11	High IEEE	High	Low
J. Cobb	Assistant Professor	FT	Ph.D.	U. T. Austin '96	1	10	7	High IEEE	High	Low
K. Cooper	Assistant Professor	FT	Ph.D.	UBC '00	0	5	5	INCOSE	High	Low
O. Daescu	Assistant Professor	FT	Ph.D.	Notre Dame '00	0	5	5	ACM	High	None
G. Dattatreya	Associate Professor	FT	Ph.D.	Indian Inst. Of Science '81	0	20	18	High IEEE	High	Low
J. Dong	Assistant Professor	FT	Ph.D.	Waterloo '02	3	3	3	High ACM, IEEE	High	Low
T. Farage	Senior Lecturer	FT	M.S.	UTD '85	6	14	14		Low	Low
A. Farago	Professor	FT	Ph.D.	Budapest '81	0	27	7	High IEEE	High	Low
G. Gupta	Professor	FT	Ph.D.	U. of North Carolina '92	0	13	5	High IEEE,ACM	High	Medium

Name	Rank	FT Or PT	Highest Degree	Institution from which Highest Degree Earned & Yr.	Years of Experience			Level of Activity (high,med,low,none) in:		
					Govt./ Ind. Practice	Total Faculty	At UTD	Prof. Society	Research	Consult/ Summer Work in industry
S. Harabagiu	Associate Professor	FT	Ph.D.	USC '97	1	7	3	IEEE, AAAI, ACM	High	Low
H. Harrison	Senior Lecturer	FT	Ph.D.	UT Austin '72	29	4	4		Low	None
V. Hatzivassiloglou	Associate Professor	FT	Ph.D.	Columbia '98			1	ACL, ISCB, AAAI	High	Low
D.T. Huynh	Professor	FT	Ph.D.	U. of Saarland Germany '78	0	23	19	High IEEE, ACM	High	Low
J. Jue	Associate Professor	FT	Ph.D.	U.C. Davis '99	0	6	6	High IEEE, ACM	High	None
S. Karrah	Senior Lecturer	FT	M.S.	Jackson State Univ. '94	8	11	5	NACADA	Low	None
L. Khan	Assistant Professor	FT	Ph.D.	USC '00	0	5	5	High IEEE	High	Low
L. King	Senior Lecturer	FT	M.S.	New Mex. St. Univ '01	30	4	4		Medium	Low
R. Krishnan	Senior Lecturer	FT	Ph.D.	Penn. State Univ. 1991	0	13	4		High	Low
R. Lacambra	Senior Lecturer	FT	M.S.	George Wash. '97	5	3	3		Low	None
J. Leubitz	Senior Lecturer	FT	M.S.	Ohio State '71	30	15	4	High ACM, Med (ISACA, PMI)	Low	High
R. Mili	Associate Professor	FT	Ph.D.	Ottawa '96	0	10	10	High IEEE,ACM	High	Low

Name	Rank	FT Or PT	Highest Degree	Institution from which Highest Degree Earned & Yr.	Years of Experience			Level of Activity (high,med,low,none) in:		
					Govt./ Ind. Practice	Total Faculty	At UTD	Prof. Society	Research	Consult/ Summer Work in industry
N. Mittal	Assistant Professor	FT	Ph.D.	UT-Austin '02	0	3	3	ACM, IEEE	High	Low
D. Moldovan	Professor	FT	Ph.D.	Columbia ‘78	3	26	4	ACM, AAAI	High	High
V. Ng	Assistant Professor	FT	Ph.D.	Cornell, ‘04	0	1	1	ACL, AAAI	High	None
S. Ntafos	Professor	FT	Ph.D.	Northwestern ‘79	0	27	26	Med. ACM	High	None
G. Ozbirn	Senior Lecturer	FT	M.S.	SMU ‘00	12	5	5		Low	Low
I. Page	Associate Professor	FT	Ph.D.	Brunel, UK ‘79	4	24	24	Medium ACM	Medium	Low
B. Prabhakaran	Associate Professor	FT	Ph.D.	IIT Madras ‘95	0	8	5	ACM, IEEE	High	Low
R. Prakash	Associate Professor	FT	Ph.D.	Ohio State ‘96	0	9	8	High ACM,IEEE	High	Low
B. Ragavachari	Professor	FT	Ph.D.	Penn. State ‘92	2	12	12	High ACM	High	Low
D. Russo	Senior Lecturer	FT	M.S.	SMU '96	19	3	3	Medium ACM	Low	High
M. Sanchez	Senior Lecturer	FT	M.S.	George Wash. ‘97	5	4	4		Low	None
K. Sarac	Assistant Professor	FT	Ph.D.	UC Santa Barbara ‘02	0	3	3	ACM, IEEE	High	None
H. Schweitzer	Associate Professor	FT	Ph.D.	Hebrew University ‘86	0	14	14	High IEEE,ACM	High	Low
E. Sha	Professor	FT	Ph.D.	Princeton ‘92	0	13	5	High ISCA, IEEE, ACM	High	None

Name	Rank	FT Or PT	Highest Degree	Institution from which Highest Degree Earned & Yr.	Years of Experience			Level of Activity (high,med,low,none) in:		
					Govt./ Ind. Practice	Total Faculty	At UTD	Prof. Society	Research	Consult/ Summer Work in industry
C. Steinhorst	Senior Lecturer	FT	M.S.	Southwestern Louisiana ‘73	25	7	5		Low	Low
H. Sudborough	Professor	PT	Ph.D.	Penn. State ‘71	0	34	19	High ACM, IEEE	High	Low
A. Sullivan	Senior Lecturer	FT	Ph.D.	UT-Dallas '81	43	3	3	Low	Low	Medium
L. Thompson	Senior Lecturer	FT	M.S.	UT-Dallas ‘98	0	4	4		Low	None
B. Thuraisingham	Professor	FT	Ph.D.	Wales ‘79	21	5	1	ACM, IEEE, BCS, AAAS, AFCEA	High	Medium
K. Truemper	Professor	FT	Ph.D.	Case Western Reserve ‘73	4	32	32	High SIAM, ORSA	High	Low
R. Uma	Assistant Professor	FT	Ph.D.	Polytechnic Univ. ‘00	0	5	5	High ACM	High	None
N. Van Ness	Senior Lecturer	FT	M.S.	Stanford ‘66	0	13	8		Low	None
S. Venkatesan	Associate Professor	FT	Ph.D.	U. of Pittsburgh ‘88	2	15	15	High IEEE	High	Medium
Y. Wang	Associate Professor	FT	Ph.D.	Saskatchavan ‘96	0	9	5	High IEEE	High	Low
W. Wei	Senior Lecturer	FT	Ph.D.	Missouri-KC ‘93	10	2	2		Low	Low
E. Wong	Associate Professor	FT	Ph.D.	Purdue ‘93	9	3	3	High IEEE	High	Low
W. Wu	Assistant Professor	FT	Ph.D.	Minnesota ‘02	0	3	3		High	None

Name	Rank	FT Or PT	Highest Degree	Institution from which Highest Degree Earned & Yr.	Years of Experience			Level of Activity (high,med,low,none) in:		
					Govt./ Ind. Practice	Total Faculty	At UTD	Prof. Society	Research	Consult/ Summer Work in industry
I.L Yen	Associate Professor	FT	Ph.D.	Houston '92	4	13	8	High IEEE	High	Low
K. Zhang	Professor	FT	Ph.D.	Brighton, UK '90	0	14	5	High IEEE, ACM	High	None
Y. Zhang	Assistant Professor	FT	Ph.D.	Arizona '02	0	3	3	ACM, IEEE	High	None
S.Q. Zheng	Professor	FT	Ph.D.	UC Santa Barbara '87	4	18	7	High IEEE,ACM	High	Medium

IV. Curriculum.

INTENT: The curriculum is consistent with the program's documented objectives. It combines technical requirements with general education requirements and electives to prepare students for a professional career in the computer field, for further study in computer science, and for functioning in modern society. The technical requirements include up-to-date coverage of basic and advanced topics in computer science as well as an emphasis on science and mathematics.

(Curriculum standards are specified in terms of semester hours of study. Thirty semester hours generally constitutes one year of full-time study and is equivalent to 45 quarter hours. A course or a specific part of a course can only be applied toward one standard.)

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

A. Title of Degree Program. Give the title of the degree program under review, as specified on the transcript and diploma:

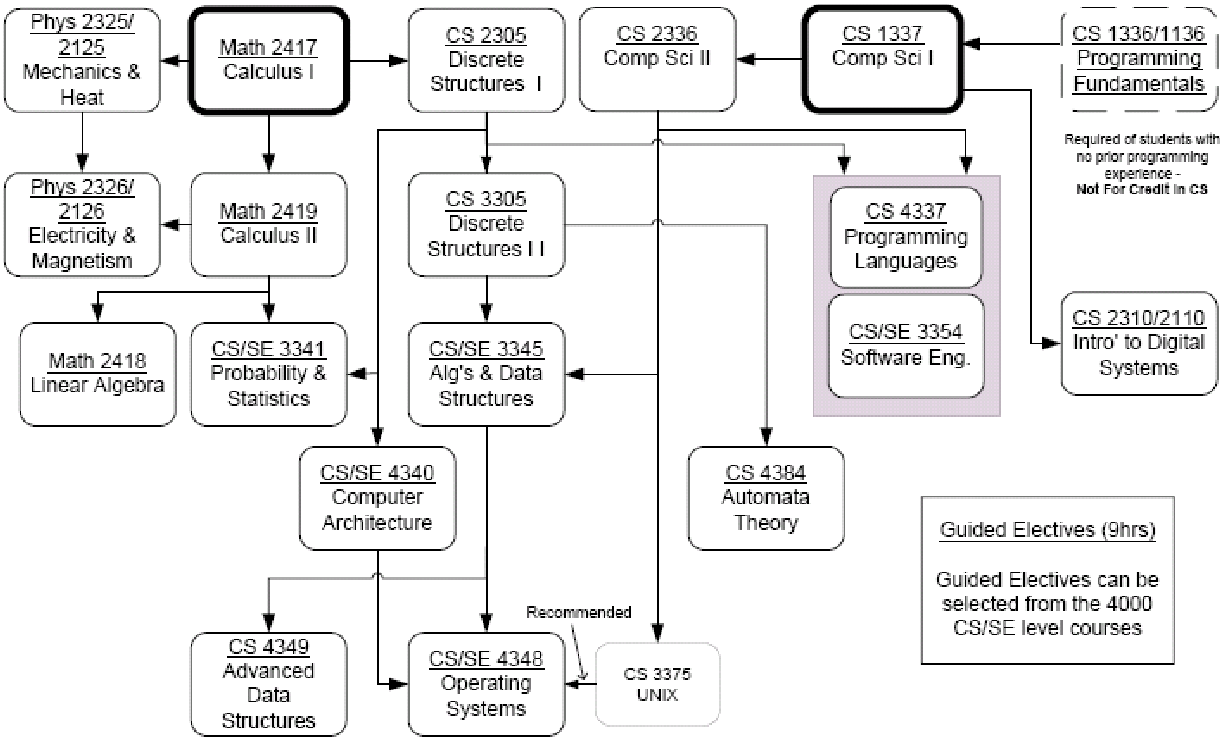
Transcript: Bachelor of Science, Computer Science

Diploma: Bachelor of Science, Computer Science

B. Credit Hour Definition. One credit hour normally means one hour of lecture or three hours of laboratory per week. One academic year normally represents from twenty-eight to thirty weeks of classes, exclusive of final examinations. Please describe below if your definitions differ from these.

No difference.

C. Prerequisite Flow Chart. Attach a flow chart showing the prerequisite structure of computer science courses required or allowed towards the major.



B.S. Computer Science Prerequisites 2004-2006.

D. Course Requirements of Curriculum (term by term and year by year)

Required and elective courses: In the tables on the following pages, List the courses in the order in which they are normally taken in the curriculum, classified in the appropriate categories. The data should clearly indicate how the program satisfies the CAC/ABET/CSAB criteria for curriculum as prescribed in the current issue of *Criteria for Accrediting Programs in Computer Science in the United States*.

Required courses: List courses by department abbreviation (Math, Chem, CS, etc.), number, title, and number of credits. Apportion the credits for each course by category.

Elective courses: Designate these courses “elective.” If an elective is restricted to a particular category, then tabulate the credit hours in that category and indicate the category in the listing, e. g. “elective—science.” In addition, be sure that you have supplied information elsewhere in this document indicating how you ensure that students take the course in the specified category (e. g. advisement, graduation check sheets, etc.). For free electives (i. e., those not restricted to a particular category), list the credits under Other. Use footnotes for any listings that require further elaboration.

Note: Individual courses may be split between or among curriculum areas if the course content justifies the split. For example, a discrete mathematics course may have some of its credits under mathematics and some under computer science. In such cases, assign credits to categories in multiples of one-half credit.

Table IV.1: Course Requirements of Curriculum 2002-2004 Catalog: Computer Science							
Year, Semester or Quarter	**Course (Department, Number, Title)**	**Category (Credit Hours)**					
		Comp. Science Core	Comp. Science Adv.	Math	Science	General Ed.	Other
1st Yr, 1st Semester	Math 2417 Calculus I			4			
	CS 1337 Computer Science I	3					
	Rhet 1302 Crit Thnkg & Writing					3	
	Rhet 1101 Oral Communication					1	
	Hst 1301 American History					3	
1st Yr, 2nd Semester	CS 2336 Computer Science II	3					
	Math 2419 Calculus II			4			
	Phys 2325 Mechanics &Heat Phys 2125 Physics Lab				4		
	Hst 2301 American History					3	
2nd Yr, 1st Semester	CS 2305 Discrete Math I	1		2			
	CS 2310/2110 Intro to Digital Syst	4					
	Phys 2326 Electromagnetism Phys 2126 Physics Lab				4		
	Govt 2301 Constitution, Found'ns					3	
	Math 2418 Linear Algebra			4			
2nd Yr, 2nd Semester	CS 3305 Discrete Math II	1		2			
	CS 3341 Probability & Statistics	1		2			
	HUMA 1301 Expl'n of Humanities					3	
	ISSS 3360 Politics & Values					3	
	Govt 2303 Inst'ns in US Econ.					3	

3rd Yr, 1st Semester	CS 3345 Alg’s & Data Str.	3					
	ECS 3390 Professional & Technical Writing	1				2	
	CS 4340 Comp Arch		3				
	AP 1301 Explor’n of the Arts					3	
	Science Elective				4		
3rd Yr, 2nd Semester	CS 3354 Software Engineering	3					
	CS 4337 Programming Lang’s		3				
	CS 4384 Automata Theory		3				
	Free Elective						3
	Free Elective						3
4th Yr, 1st Semester	CS 4348 Operating Systems		3				
	CS Guided Elective		3				
	CS Guided Elective		3				
	Advanced Free Elective						3
	Free Elective						3
4th Yr, 2nd Semester	CS 4349 Advanced Data Str.		3				
	CS Guided Elective		3				
	Free Elective					3	
	Free Elective					3	
	Advanced Free Elective						3
Totals		20	24	18	12	33	15

Note: 6 hours of the free electives need to be in broadening fields/humanities. Thus, a student ends up taking 33 hours of such classes. Note that transfer students do not take Rhet 1101, so they will take only 32 hours of courses in broadening fields/humanities which is still more than the 30 hours required for ABET accreditation.

Table IV.2 : Basic Core Courses:

Course (Dept., Number, Title)	Category (credit hours)					
	Theo. Found.	Algorithms	Data Structures	Software Design	Prog. Concepts	Computer Arch.
CS 1337 Computer Science I				**0.5**	**2.5**	
C3 2305 Discrete Math I	**1**					
CS 2310/2110 Intro to Digital Systems	**1**					**3**
CS 2336 Computer Science II			**1**	**1**	**1**	
CS 3305 Discrete Math II	**1**					
CS 3341 Probability and Statistics	**1**					
CS 3345 Algorithm Analysis and Data Structures	**0.5**	**1**	**1**	**0.5**		
CS 3354 Software Engineering	**1**			**2**		
ECS 3390 Technical and Professional Communication **(1 Cr. towards core)**	*	*	*	*	*	*
Total Core	5.5	1	2	4	3.5	3
* allocation varies by student (projects)						

Total credits (incl. 1 Cr of ECS 3390): <u>20</u>

Table IV.3: Advanced Core Courses (excluding guided electives):

Course (Dept., Number, Title)	Category (credit hours)					
	Theo. Found.	Algorithms	Data Structures	Software Design	Prog. Concepts	Computer Arch.
CS 4337 Organization of Programming Languages	0.5		0.5		2	
CS 4340 Computer Architecture						3
CS 4348 Operating Systems Concepts	0.5	0.5		1.5	0.5	
CS 4349 Advanced Data Structures and Algorithms		2	1			
CS 4384 Automata Theory	3					
Total Core	4	2.5	1.5	1.5	2.5	3

Total credits: 15

In addition to the above required courses, students must take at least 9 semester hours of CS guided electives, which are 4000 level CS courses approved by the CS advisor. The following courses may be used as guided electives without explicit approval of an advisor.

Table IV.4: CS Elective Courses.

Course (Dept., Number, Title)	Category (credit hours)					
	Theo. Found.	Algorithms	Data Structures	Software Design	Prog. Concepts	Computer Arch.
CS 4334 Numerical Analysis	1	1		1		
CS 4336 Advanced Java Programming			1	1	1	
CS 4347 Database Systems	1		1	1		
CS 4361 Computer Graphics	0.5	0.5		2		
CS 4365 Artificial Intelligence	1	1		1		
CS 4376 Object-Oriented Programming Systems				1.5	1.5	
CS 4380 Senior Design Project				1.5	1.5	
CS 4386 Compiler Design	1	1		1		

CS 4390 Computer Networks	0.5	1	0.5	1		
CS 4391 Computer Vision	0.5	1.0	0.5	1		
CS 4392 Computer Animation		1	1	1		
CS 4393 Computer and N/W Security				2.5		0.5
CS 4394 Impl. Of Modern OS		0.5	0.5	1.5	0.5	
CS 4396 Networking Lab	0.5	0.5	1	1		
CS 4397 Embedded Systems		0.5		2.5		
EE 4325 Introduction to VLSI Design						3
EE 4420 Microprocessor Syst Design						3
SE 4351 Requirements Engineering				3		
SE 4352 SW Architecture and Design				3		
SE 4367 Software Testing, Verification, Validation and Quality Assurance				3		
SE 4485 Software Engineering Project				3		

The *Intent* stated at the beginning of this section must be met in order for a program to be deemed accreditable. One way to meet the *Intent* of this criterion is to satisfy each one of the Standards listed below. To do this, answer the questions associated with the Standards. If one or more Standards are not satisfied, it is incumbent upon the institution to demonstrate and document clearly and unequivocally how the *Intent* is met in some alternative fashion.

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which pro gram is being described.

Standard IV-1. The curriculum must include at least 40 semester hours of up-to-date study in computer science topics.

1. If it is not obvious from the above tables that the curriculum includes at least 40 semester hours (60 quarter hours) of computer science topics, please explain.

The curriculum includes 44 hours of Computer Science core classes (Table IV.1).

Standard IV-2. The curriculum must contain at least 30 semester hours of study in mathematics and science as specified below under Mathematics and Science.

2. If it is not obvious from the above tables that the curriculum includes at least 30 semester hours (45 quarter hours) of study in mathematics and science, please explain.

The curriculum includes 18 hours of Mathematics and 12 hours of Science (Table IV.1).

Standard IV-3. The curriculum must include at least 30 semester hours of study in humanities, social sciences, arts and other disciplines that serve to broaden the background of the student.

3. If it is not obvious from the above tables that the curriculum includes at least 30 semester hours (45 quarter hours) of study in humanities, social sciences, arts, and other disciplines that serve to broaden the background of the student, please explain.

   We have 27 semester hours required outside the free electives. Students are required to take 15 semester hours of free electives and 9 semester hours of advanced electives. Advisors have encouraged students in the past to broaden their education by taking at least some of their electives in the Humanities. In the 2004-06 catalog, what has been common practice is made an explicit requirement (for at least 6 hours of electives in "broadening areas") to ensure that the requirement of 30 hours is met (see footnote 7 in the "Degree Plan" included in Appendix IX).

Standard IV-4. The curriculum must be consistent with the documented objectives of the program.

Our curriculum is indeed consistent with our educational objectives EO1-EO5. The computer science core classes, the Math and Science requirements, and the Computer Science electives ensure that objective EO1 is met. Our class ECS 3390 as well as emphasis on projects and project reports ensures that objective EO2 is met. Computer science core and elective classes ensure that objective EO3 is met. Our program emphasis on teaching general principles (rather than on teaching popular tools of the moment), our emphasis on laying down strong theoretical foundations and developing strong programming skills ensures that objective EO4 is met. The required class ISSS 3360 and added emphases in other Computer Science classes ensures that objective EO5 is met

Standard IV-5. All students must take a broad-based core of fundamental computer science material consisting of at least 16 semester hours.

4. If it is not obvious from the above tables that the curriculum includes a broad-based core of fundamental computer science material consisting of at least 16 semester hours (24 quarter hours), please explain.

The basic core in Table IV.1 includes 20 hours.

Standard IV-6. The core materials must provide basic coverage of algorithms, data structures, software design, concepts of programming languages and computer organization and architecture.

5. The core materials must provide basic coverage of the following five areas. Please indicate below the approximate number of hours in the core devoted to each topic. (This material can be gathered from your course descriptions, but it will ease the job for the visiting team if you do this in advance.)

   Algorithms _3.5____,
   Data Structures _3.5____,
   Software Design __5.5___,
   Concepts of Programming Languages __6___,
   Computer Organization and Architecture __6___.

The numbers above include the core classes in Tables IV.2 and IV.3 only. The total coverage of these areas in the curriculum depends on the student's choice of guided electives (coverage is at least as much as shown above).

Standard IV-7. Theoretical foundations, problem analysis, and solution design must be stressed within the program's core materials.

6. The following areas must be stressed within the program's core materials. Indicate the course numbers of courses embodying a significant portion of these areas:

Theoretical Foundations:	CS 2305, CS 2310/2110, CS 2336, CS 3345, CS 4349, CS 4384
Problem Analysis:	CS 1337, CS 2305, CS 2310/2110, CS 2336, CS 3305, CS 3345, CS 3354, CS 4337, CS 4340, CS 4348, CS 4349, CS 4384
Solution Design:	CS 1337, CS 2310/2110, CS 2336,CS 3345, CS 3354, CS 4340, CS 4348, CS 4349.

Standard IV-8. Students must be exposed to a variety of programming languages and systems and must become proficient in at least one higher-level language.

7. Typically, to what programming languages and operating systems are your students exposed?

The Programming sequence was recently switched from C/C++ to Java. However, classes like Operating Systems and Networks still require knowledge of C/C++ (the students may take CS 3335 or pick it up on their own; also some Instructors provide tutorials). By graduation, students should be proficient in both Java and C/C++ programming as well as both the Unix and Windows Operating Systems. Students also pick up a good working knowledge of ML (functional programming language) and Prolog (logic programming language) when they take CS 4337 (Programming Languages).

8. In what computer language(s) do your students become proficient?

   Java, C/C++, Assembly Language (see above).

Standard IV-9. All students must take at least 16 semester hours of advanced course work in computer science that provides breadth and builds on the core to provide depth.

9. If it is not obvious from the tables above that your students take at least 16 semester hours (24 quarter hours) of advanced computer science, please explain.

Table IV.1 includes 24 hours of advanced Computer Science topics (15 core hours and 9 hours of guided electives.

10. List below the advanced areas in which your students may study. Make clear by your use of "and" and "or" and parentheses which areas are required and which may be chosen from (e. g., A and two of (B or C or D)).

Depending of the choice of Guided Electives and free electives, a student could concentrate on Software Engineering (CS/SE 3354, SE 4351, SE 4352, SE 4367, SE 4485), Compilers (CS 4337, CS 4384, CS 4386), Embedded Systems (CS 4348, CS 4394*, CS 4397*), Computer Imaging (CS 4361, CS 4391*, CS 4392*), Networks (CS 4390, CS 4393*, CS 4396*). The courses marked with an asterisk are new classes in the 2004-06 catalog. Most students will select one of these options but it is feasible to get good coverage of two areas while still allowing for a broad educational experience.

Standard IV-10. The curriculum must include at least 15 semester hours of mathematics.

11. If it is not obvious from the tables above that your students take at least 15 semester hours (23 quarter hours) of mathematics , please explain.

Table IV.1 includes 18 hours of Mathematics.

Standard IV-11. Course work in mathematics must include discrete mathematics, differential and integral calculus, and probability and statistics.

12. If it is not obvious from course titles in the above tables, then explain below which required courses contain discrete mathematics, differential and integral calculus, and probability and statistics.

The 18 hours of Mathematics include 4 in Discrete Mathematics, 4 in Linear Algebra, 8 in Calculus, and 2 in Probability and Statistics. The probability and statistics course content has been considerably revised to streamline the material to the needs of the computer science students. The revised content is being taught since Fall 2004 semester.

Standard IV-12. The curriculum must include at least 12 semester hours of science.

13. If it is not obvious from the tables above that your students take at least 12 semester hours (18 quarter hours) of science, please explain.

Table IV.1 includes 8 hours of Physics (with lab) and a 4 hour Science Elective for a total of 12 hours.

Standard IV-13. Course work in science must include the equivalent of a two-semester sequence in a laboratory science for science or engineering majors.

14. If it is not obvious from the tables above and from course descriptions and/or your catalog that the science requirement includes a full year (two-semester or three-quarter) sequence in a laboratory science for science and engineering majors, please explain.

This is provided by the 8 hour Physics sequence (with 2 laboratory hours)

Standard IV-14. Science course work additional to that specified in Standard IV-13 must be in science courses or courses that enhance the student's ability to apply the scientific method.

15. If it is not obvious from the tables above and from course descriptions and/or your catalog that the remainder of the science requirement is met with science courses or courses that enhance the student's abilities in the application of the scientific method, please explain. (Mathematics, statistics, and courses normally considered part of the computer science discipline should not be included here).

4 semester hours of science electives are counted under Science core.

Standard IV-15. The oral communications skills of the student must be developed and applied in the program.

Standard IV-16. The written communications skills of the student must be developed and applied in the program.

16. Each student's oral and written communications skills must be developed and applied in the program, i. e., in courses required for the major. This information should be included in course descriptions; please give course numbers below.

Most General Education courses contain a significant writing component. Thus, every program on campus is involved in helping students to acquire and strengthen this competence.
The primary unit responsible for development of communication skills at the lower division is the School of Arts and Humanities through the rhetoric program. For the advanced writing requirement, the School of Engineering and Computer Science is responsible. In addition, students are encouraged to develop their oral skills through participation in ACM, IEEE student chapter activities.

At the lower level, all engineering students are required to complete a course in English rhetoric, RHET 1302 (Critical Thinking and Writing), with a heavy writing requirement. At the upper level, all UT Dallas students are required to complete an advanced writing requirement in their discipline. In Computer Science, students take CS 3390 (Professional and Technical Communication). This course utilizes an integrated approach to writing and speaking for the technical profession. The writing component focuses on writing professional quality technical documents such as proposals, memos, abstracts, reports and letters. The oral communication part of the course focuses on planning, developing, and delivering dynamic, informative and persuasive presentations.

Computer Science courses require written work in the form of program documentation, term papers, and written homework. Several courses require oral communication. For example, software design projects in courses may require students to give individual or team presentations of their designs. Some senior level courses have a term research paper as a requirement. Faculty members are encouraged to require writing assignments in all courses. The grading of laboratories and lecture reports reflects the quality of writing, and corrective action usually requires re-writing assignments until the presentation is acceptable.

Standard IV-17. There must be sufficient coverage of social and ethical implications of computing to give students an understanding of a broad range of issues in this area.

17. Social and ethical implications of computing must be covered in the program. This information should be included in course descriptions; please give course numbers below.

ISSS 3360 and ECS 3390 cover contemporary issues associated with computing. CS students are encouraged to take CS 3385 (Ethics, Law, Society, and Computing) as an elective. Ethical and social impacts of computing are addressed in the introductory core courses CS 1337 and CS 2336 as part of developing an understanding of the role of computers and especially that of computer sciences. Exposure continues to varying extends throughout the curriculum and especially in classes like CS 3354 (Software Engineering), CS 4347 (which addresses the individual and organizational concerns regarding accuracy, privacy and integrity of data), CS 4348 and CS 4390 (which address the security and privacy issues arising in operating systems and computer networks), and the newly added classes (e.g., CS 4393-Computer and Network Security).

The fact that our student body includes significant numbers that work in local industry (full-time or in the Co-op programs) serves to raise ethical and social issues in many classroom discussions.

E. Course Descriptions.

For each required or elective computer science course that can be counted for credit in the curriculum being reviewed for accreditation, include a two-page or three-page course outline at this point in the self-study. If your documentation does not exactly follow this format, be sure that all of the indicated information (if applicable) is present, and please in any case adhere to a common format for all course descriptions.

Note: The outline format calls for information on the content of the course in the areas of computer science theory, communications skills development and application, social and ethical implications of computing, and problem analysis and solution design experiences. This is not intended to suggest that every course must have some coverage of each of these topics. For a given course, please include the information from a listed area only if the course has significant content in that specific area.

In addition, similar outlines should also be included for required mathematics and science courses taken by computer science students.

The course outline for each required or elective computer science course must also be included in a display of course materials that is available for study at all times during the program evaluation site visit. The course material display must include at least the following for each computer science course that can be counted in the computer science segment of the curriculum being evaluated.

1. Textbook and other required material (e.g., manuals, reference booklets, standards documents, and so forth)
2. Syllabus and course policies
3. A complete set of assignments, tests, and important handouts
4. Samples of graded student work on all assignments, written reports and other documents, and tests. Examples of excellent, satisfactory, and poor student work should be included.

2. If some of the above documentation is online (e. g., in an instructor's web site), please indicate this, and have a computer available at or near the course displays so that the team can view it. Please give here the URL(s) for accessing any such materials:

The Undergraduate catalog includes course descriptions and can be found at:

URL : http://www.utdallas.edu/student/catalog/undergrad04/ugprograms/ecs-cs.html

The current Undergraudate Catalog (CS Section) is also included in Appendix VIII.

COURSE DESCRIPTIONS

All course descriptions are included in **Appendix XI**.

V. Laboratories and Computing Facilities

INTENT: Laboratories and computing facilities are available, accessible, and adequately supported to enable students to complete their course work and to support faculty teaching needs and scholarly activities.

The ***Intent*** must be met in order for a program to be deemed accreditable. One way to meet the ***Intent*** of this criterion is to satisfy each one of the **Standards** listed below. To do this, answer the questions associated with the **Standards**. If one or more **Standards** are not satisfied, it is incumbent upon the institution to demonstrate and document clearly and unequivocally how the ***Intent*** is met in some alternative fashion.

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

> In Section VI we will ask you to describe laboratory equipment planning and acquisition processes. Please do not repeat any of that information here; simply refer ahead to that section if necessary to avoid duplication.

A. Computer facilities available for use in computer science programs. Describe the computer facilities available for use in programs in computer science.

1. Describe the computing facilities used for instruction. Indicate the types of software available in each category. Specify any limitations that impact the quality of the educational experience.

Institutional facilities:

ECSS 2.103 is a large open lab with over 100 PCs for general use. Most projects for the beginning programming classes are done there. Portions can be reserved for specific classes to carry out hands-on instruction. Most software needed for classes are available at this lab including Compilers and Language Development Environments, several tools from Rational, x-SUDS from Telcordia (testing tools), etc. Students also have access to large computer servers that the department owns, and that can be accessed through the workstations in the labs.

Departmental facilities:

ECSN 3.112, 3.118, 3,120 are shared with Electrical Engineering; they are used to support classes in Digital Systems and Computer Architecture. Circuit design boards and software support are provided.

Several other laboratories in ECSS are dedicated to instruction and provide advanced environments for specific needs (e.g., UNIX workstations and dual-boot PCs to support CS 3375 and Operating Systems projects, Advanced Software Engineering tools to support senior Software Engineering classes, DSP labs, Labs dedicated to Operating Systems and Networking classes).

2. Are there any labs, courses, or policies that require two or more students to share a lab station? _________ If the answer is yes, please describe the situation(s) involved.

No; at present there is general satisfaction with the computing facilities.

3. Briefly describe the laboratory equipment planning, acquisition, and maintenance processes and their adequacy. Include discussion of these topics for university-wide facilities available to all students (if used by your majors), your own laboratories and equipment (if applicable), and facilities controlled by other departments and/or schools (if used by your majors). Discuss how you assess the adequacy of your laboratory and computing support. Please attach documentation (e. g., inventories, equipment replacement plans, etc.) to this report.

The equipment in ECSS is rather new since the building is only 2 years old. When the current building was built in 2002, $600,000 was allocated for buying equipment for the various Labs that were set up. These funds were used to buy the equipment in the Labs described previously. The CS computing committee and the CS technical staff have developed and are implementing plans to upgrade equipment on a 3-4 year staggered cycle.

Standard V-1. Each student must have adequate and reasonable access to the systems needed for each course.

B. Student Access. Each student must have adequate and reasonable access to the systems needed for each course. State the hours the various facilities are open. State whether students have access from dormitories or off campus by direct access, modem, etc., and describe this access quantitatively.

The general use lab (ECSS 2.103) is open 18 hours a day (the lab is supervised by teaching assistants to prevent theft, vandalism and the availability of the assistants is the only reason the lab is not always open). The other labs in the building are accessible though computer controlled entry and are available anytime to students that are authorized (by virtue of the classes they are registered for) to use them. UTD provides several remote access options (RNA, Pipeline); wireless network access is available in most buildings and in student housing.

Standard V-2. Documentation for hardware and software must be readily accessible to faculty and students.

C. Documentation. Describe documentation for hardware and software systems available to students and faculty in the computer science program. Explain how students and faculty have adequate and timely access to the documentation.

Documentation is usually available online with hard-copy manuals as needed (in labs or on reserve in the Library).

Standard V-3. All faculty members must have access to adequate computing facilities for class preparation and for scholarly activities.

D. Faculty Access. Describe the computing facilities available to faculty for class preparation and for scholarly activities and research. Include specifics regarding resources in faculty members' offices.

Faculty have up-to-date PCs and/or Sun Workstations in their offices. In addition, most research active faculty members have set up labs that house any where from 6 to 12 computers that they use for research. In addition, the department has several large and powerful servers that can be accessed through workstations. A comprehensive list of research and teaching labs is shown below:

List of Departmental facilities:

- Graduate Students Open Lab
- Computer Software Engineering Open Lab
- CS Tutoring Lab
- Embedded Software Center
- Intervoice Center for Conversational Technologies (Human Language Technology Research Institute)
- Center for Search Engines and Web Technologies (Human Language Technology Research Institute)
- Center for Text Mining (Human Language Technology Research Institute)
- Center for Basic Research in Natural Language Processing (Human Language Technology Research Institute)
- Center for Emerging Natural Language Applications (Human Language Technology Research Institute)
- Security Analysis and Information Assurance Lab/Cyber Security Center

Digital Forensics and Emergency Preparedness Institute
Distributed Systems and Internet Computing Lab
Applied Logic, Programming-Languages and Systems Lab (ALPS)
Software and Information Visualization Lab
DSP and Communications Lab
Wireless information systems Lab
Multimedia Distance Learning Lab
Parallel Computation Lab
Artificial Intelligence Lab
CAD and Visualization Lab
Database Laboratory
Telecommunications Lab
Computer Vision and Multimedia Systems Lab
Telecommunications and SE Lab
Resource Allocation and Scheduling Lab
Laboratory of Advanced Computer and Network Architectures
Advanced Networking and Dependent Systems Laboratory
Multimedia Systems and Networking Lab
Software Technology Advanced Research
Compiler and Architecture Research Lab
NET Lab - Scalable Network Engineering Techniques Laboratory
Visual Computing
Formal Method Lab

Software Architecture Lab
Advanced Network Research Lab
Advanced Computation Lab
Requirements Engineering Lab

Standard V-4. There must be adequate support personnel to install and maintain the laboratories and computing facilities.

E. Support Personnel.

1. What support personnel are available to install, maintain, and manage departmental hardware, software, and networks?

The Department of Computer Science employees three technical support staff (Brian Nelson, Harold Clark, and Cody Crudgington). They are assisted by several students assigned to them as Assistants or employed on an hourly basis). An additional technical support staff member (Mark Hittenger) maintains machines in the Human Language Technology Research Institute (HLTRI) and is supported by external grant funds.

2. Is this level of support adequate? __________ If not, describe the limitations.

Technical support is deemed more than adequate at this time.

3. Are any faculty members expected to provide significant hardware, network, or software support? If so, describe this expectation including how such expectations are addressed in evaluation, tenure, promotion, and merit pay decisions, and indicate what, if any, released time is awarded for this effort.

No faculty is directly involved in providing hardware/software support. Faculty members serving in the Equipment Committee are involved in planning and coordinating activities with the technical staff. Service in departmental, university committees is expected of all faculty (with reduced expectations from junior faculty) and is included in annual reviews under the "service" component (together with research and teaching).

Standard V-5. Instructional assistance must be provided for the laboratories and computing facilities.

F. Instructional Support. Describe the nature and extent of instructional support available to students in the laboratories.

Instructional support is needed mostly for the beginning classes and is provided by Teaching Assistants. Senior mentors (undergraduates working 10 hours a week) are employed to assist with the CS 1336 lab and that has worked very well. Assistance for the more advanced classes is provided by the Teaching Assistant assigned to the class and the instructor as needed.

VI. Institutional Support and Financial Resources.

INTENT: The institution's support for the program and the financial resources available to the program are sufficient to provide an environment in which the program can achieve its objectives. Support and resources are sufficient to provide assurance that an accredited program will retain its strength throughout the period of accreditation.

The ***Intent*** must be met in order for a program to be deemed accreditable. One way to meet the ***Intent*** of this criterion is to satisfy each one of the **Standards** listed below. To do this, answer the questions associated with the **Standards**. If one or more **Standards** are not satisfied, it is incumbent upon the institution to demonstrate and document clearly and unequivocally how the ***Intent*** is met in some alternative fashion.

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

Standard VI-1. Support for faculty must be sufficient to enable the program to attract and retain high-quality faculty capable of supporting the program's objectives.

UT Dallas Computer Science has a distinguished faculty. Faculty members published more than 200 scholarly papers in 2004, served in editorials boards of more than 30 journals, helped in the organization of a large number of conferences and workshops, made more than 100 trips to attend scholarly conferences and workshops, and raised millions of dollars in extra-mural research funding. All this has become possible only because of high level of support provided by the administration, both materially and morally. Salaries provided to the faculty members are highly competitive, so are the start-up packages. Recent faculty hires have been given start-up packages in 6 figures. The administration provides very high level of support for cost matching to help faculty members attract extra-mural funding. With the JSRE initiative funds, a large number of fellowships for research students, travel support, seed money for projects have been provided to make the environment extremely attractive to faculty members. In the 2005 hiring season, the department interviewed 10 candidates, made 5 offers, out of which 3 accepted. This points to the attractiveness of UT Dallas Computer Science as a department.

Standard VI-2. There must be sufficient support and financial resources to allow all faculty members to attend national technical meetings with sufficient frequency to maintain competence as teachers and scholars.

There are adequate financial resources. UTD Computer Science faculty members made close to than 150 trips in 2004 alone. These trips were financed by external research grants (which are highly encouraged by the School/Department administration) as well as the School's and Department's internal funds. In fact, from 2004 onwards, the Dean has promised to support one trip per year for non-tenured track faculty members (who generally do not have any extramural funding to support their own travel).

Standard VI-3. There must be support and recognition of scholarly activities.

A. One evidence of the long-term stability of a program is its ability to both attract and retain high quality faculty. Describe how your program does this. Some topics the description might address are sabbatical and other leave programs, salaries, benefits, teaching loads, support for and recognition of scholarly activity (including financial support for attendance at professional meetings), departmental and institutional ambiance, etc. Give counts of the total number of faculty and the number of resignations, retirements, and new hires for each of the last five years. Indicate whether there are significant problems attracting and retaining faculty, and if so, the causes.

The significant increase in the size of tenured/tenure-track faculty is shown in the table below.

Year	Total Faculty	Resignations	Retirements	New Hires
1999-2000	22	0	0	12
2000-01	34	2	0	3
2001-02	35	1	0	6
2002-03	40	1	0	0
2003-04	39	1	0	3
2004-05	41	1	0	4
2005-2006	45	n/a	n/a	n/a

In general, we have had very successful faculty searches over the past few years (tenure-track faculty size doubled in the past 5 years). Also, the quality has increased due to the attractiveness of the DFW Metroplex, the potential of the Department of Computer Science, and general market conditions. Retention has not been a problem. In addition to the tenure track faculty, we have added a net of 7 senior lecturers over the past 5 years.

B. Summarize the professional activities of your faculty, attendance at meetings, university and professional honors won by individuals, etc. Just summarize here; details should appear in individual faculty vitas.

Computer Science faculty are very active in research. Its faculty publish more than 200 papers every year. They serve in editorial boards of more than 30 journals, and participate in program committees of numerous conference both as members as well as chairs/co-chair. Faculty members are also active in organizing research conferences, workshops and summer schools. Our faculty frequently serve in NSF review panels and are called upon by the industry for expert advise. The annual research budget of the department is in the millions of dollars. Five of our faculty members currently hold the highly prestigious NSF CAREER award.

Standard VI-4. There must be office support consistent with the type of program, level of scholarly activity, and needs of the faculty members.

C. Briefly describe the level and adequacy of office support. The description should address secretarial support, office equipment, and the total group supported by this equipment and staff.

The Department of Computer Science currently employees 8 office support staff, who are assisted by 3 to 4 work-study students There are four high-end copiers and other office equipment.   The level of support is considered excellent.

Standard VI-5. Adequate time must be assigned for the administration of the program.

D. Describe the adequacy of the time assigned for the administration of the program.

The Computer Science program is administered by the Department Head (D.T. Huynh), with assistance from an Associate Head (Gupta), and an Assistant Head (Raghavachari); they allocate roughly 50% of their time to administrative duties. The current Associate Dean for Undergraduate Education (Ntafos), as the Software Engineering Program Head, also helps in administration.

Standard VI-6. Upper levels of administration must provide the program with the resources and atmosphere to function effectively with the rest of the institution.

E. Describe the adequacy of the resources and the atmosphere provided by the upper administration for the program to function effectively with the rest of the institution.

The mission of the University of Texas at Dallas emphasizes education and research in Engineering, Science, Technology and Management.   In accordance with this mission, the School of Engineering and Computer Science has the full support of the upper administration and the University in general as it is broadly accepted that the future of the University is closely tied to the success of the School of Engineering and Computer Science.   Although UTD has faced budget shortfalls in the past few years (including a hiring freeze), the Department of Computer Science has had continued support which enabled continued hiring of high quality faculty.   The Department of Computer Science works closely with the Department of Electrical Engineering in administering Interdisciplinary programs in Telecommunications Engineering (the first in the country) and Computer Engineering.   It also has worked closely with the Department of Mathematical Sciences (Mathematics sequence), the School of Arts and Humanities (ECS 3390, Advanced Technical Communication), and collaborations are under way with Geosciences (new program on GIS and Spatial Data Mining), Mathematical Sciences and Biology (new Bio-informatics/Biotechnology programs). A new collaborative program in “Bioengineering” involving the School of Engineering and Computer Science, the School of Natural Sciences, and UT Southwestern Medical School is in the planning stage. Similarly, a new BS degree in Computer Engineering jointly offered by the Computer Science department and the Electrical Engineering department is in the planning stage.

As stated, the upper administration is highly supportive of the Computer Science department. A significant amount of JSRE Initiative funds have been spent on Computer Science faculty and students. The administration has provided travel support, matching funds for grants, fellowship and research assistantships to help faculty recruit students for their research, etc. For example, the Dean has pledged to pay for travel to one conference per year for non-tenure track faculty members to help in their professional development.

F. Positive and constructive leadership at the college/school level and within the program's department are especially important to the program's quality. Evaluate this leadership and the interaction between these levels of administration.

Dr. Robert Helms (former Professor at Stanford, Corporate Vice President at Texas Instruments, President and CEO of International Sematech) joined UT-Dallas as the Dean of the School of Engineering and Computer Science in June 2003. He had a major role in the realization of the 5-year $300 million initiative known as the JSRE Initiative (now rechristened Jonsson School Research Excellence Initiative) and is the author and driving force of the "top 50 in 5" vision for the School.

Dung T. Huynh has been the Head of the Computer Science Department since 1997. During this period, the Department has more than doubled its tenure-track faculty; integrating so many new faculty in the department in such a short time has been a major task. Student enrollment have also increased dramatically making the Department one of the largest in the nation.

What the Department, School have accomplished so far, would not be possible without strong, positive, constructive leadership, close collaboration, and faculty involvement; what we hope to accomplish in the next 5 years will require even more of the same.

Standard VI-7. Resources must be provided to acquire and maintain laboratory facilities that meet the needs of the program.

G. Laboratory and Computing Resources. Briefly describe the resources available for the program to acquire and maintain laboratory facilities. Include information on how the institution determines the adequacy of the resources.

The move to the new South wing in 2002 has provided the Department of Computer Science with plentiful laboratory facilities which should be adequate to accommodate the planned growth in the next couple of years; the addition of the new Engineering and Science Laboratory building (expected in 2006) will allow accommodation of the growth envisioned in the JSRE Initiative. The Department's Equipment Committee regularly reviews space and upgrade needs and makes recommendations to the Department Head.

Standard VI-8. Resources must be provided to support library and related information retrieval facilities that meet the needs of the program.

H. Library Resources. Briefly describe the resources available for the support of the library and related information retrieval facilities. Include information on how the institution determines the adequacy of the resources.

Library facilities and support for the library are discussed in the next section.

Standard VI-9. There must be evidence that the institutional support and financial resources will remain in place throughout the period of accreditation.

I. Discuss and show evidence of continuity of institutional support for the program in the past, and problems that have existed or are anticipated in this area, if any.

The UTD mission statement and the JSRE Initiative offer clear evidence that financial resourses will remain in place for the next 5 years.

VII. Institutional Facilities

INTENT: Institutional facilities, including the library, other electronic information retrieval systems, computer networks, classrooms, and offices, are adequate to support the objectives of the program.

The ***Intent*** must be met in order for a program to be deemed accreditable. One way to meet the ***Intent*** of this criterion is to satisfy each one of the **Standards** listed below. To do this, answer the questions associated with the **Standards**. If one or more **Standards** are not satisfied, it is incumbent upon the institution to demonstrate and document clearly and unequivocally how the ***Intent*** is met in some alternative fashion.

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

A. Library

Standard VII-1. The library that serves the computer science program must be adequately staffed with professional librarians and support personnel.

1. Assess the staffing of the library (or libraries) that serve the computer science program. Are there adequate professional librarians and support personnel? Supply documentation if possible.

The University of Texas at Dallas has three libraries to meet the informational needs of students, faculty and staff. The main library is the McDermott Library; supplemental libraries include the Callier Library, which focuses on the needs of the Callier Center for Communication Disorders, and the Engineering Reading Room, which supports the engineering programs in the Erik Jonsson School of Engineering and Computer Science. A branch library at the new building for the School of Management opened recently and focuses on supporting the needs of that School.

The mission of the Eugene McDermott Library Reference and Information Services Department is to provide the faculty, students, and staff of the University of Texas at Dallas with the information necessary to support instruction and research.

The Reference and Information Services Department provides a range of services to library users. Librarians and support staff handle over 100,000 questions per year at two service desks. In addition, the Reference librarians provide answers to reference questions submitted on a departmental WWW page.

The staff is actively involved in the overall educational goals of the University through its instructional and service approach to the provision of information. Professional librarians have created an instruction program providing basic and advanced assistance in conducting library research. Faculty members can request customized instruction to support a particular project or to familiarize students with research tools in a discipline, including the library catalog, electronic periodical indexes, the Internet, reference sources, and government publications.

The library building is open for study and access to materials in the open stacks 102 hours per week:

Library Building Hours:

Mon-Fri	8:00 am - midnight
Sat	9:00 am – 8:00 pm
Sun	1:00 pm - midnight

Reference Desk Hours:

Mon - Thurs	9:00 am – 10:00 pm
Fri	9:00 am – 8:00 pm
Sat	10:00 am – 7:00 pm
Sun	1:00 pm – 10:00 pm

See the resume of the Engineering/Computer Science librarian on the following page.

RESUME: Marjorie Henderson, Senior Librarian

Engineering/Computer Science
Erik Jonsson School of Engineering and Computer Science University of Texas at Dallas

QUALIFICATIONS
Thirty years experience in establishing specialized information centers/ libraries in diverse disciplines utilizing current and advanced technologies in information research and retrieval.

PROFESSIONAL EXPERIENCE
1994-Present University of Texas at Dallas Erik Jonsson School of Engineering and Computer Science

1987-Present University of Texas at Dallas McDermott Library Erik Jonsson School of Engineering and Computer Science

1974-1987 Xerox Corporation Office Products Division, Dallas, Texas

Established library/information services for new Xerox division pioneering office automation and advanced electronic and telecommunications engineering products and their marketing.

Established graduate student grant program with Schools of Library and Information Science at University of North Texas, Texas Women's University and East Texas State University. Extended the program to the high school level with the Dallas Independent School Business Management Center.

CAREER HIGHLIGHTS PRE 1974
Managed/organized/restructured and served in a variety of university/industry organizations in such fields as astronomy/astrophysics; marine technology/ oceanography; geophysics; electron physics and optical physics; helicopter engineering; transportation and traffic engineering; communications technology

PAPERS
How to Manage the Information Resource: a Xerox Case Study
Computerized Typesetting: Some Aspects, Techniques and Trends
Xerox Office Products Division Site Services Handbook

EDUCATION
AB, University of California, Berkeley, California
MLS, University of Southern California, Los Angeles, California

ASSOCIATIONS/ORGANIZATIONS
American Society for Information Science
Altrusa International, Inc, Richardson, Texas
Leadership Richardson, Class VI, 1991-92

Standard VII-2. The library's technical collection must include up-to-date textbooks, reference works, and publications of professional and research organizations such as the ACM and the IEEE Computer Society.

2. Assess the adequacy of the library's technical collection and of the budget for subscriptions as well as new acquisitions. The library must contain up-to-date textbooks, reference works, and publications of professional and research organizations such as the ACM and the IEEE Computer Society. It should also contain representative trade journals. Supply documentation, if possible. Assess the process by which faculty may request the library to order books or subscriptions.

Per Dr. Larry Sall, Director of the Libraries, books are and will remain central to the library's mission. As a result, the library staff constantly monitors book selections and seeks input from faculty. The library subscribes to a large number of Computer Science journals, a majority of them published by ACM and IEEE. The library also subscribes to a large number of electronic journals. Of particular significance for the computer science department are the ACM and IEEE digital libraries, Elsevier's Science Direct, Kluwer Online, Springer LINK, and Wiley Interscience.

Table VII-2.1 indicates the number acquisitions made in the last three years while table VII-2.2 provides insight on library's expenditures for the last three years.

	COLLECTION RESOURCES 2002-2003 TITLES New Acquisitions		COLLECTION RESOURCES 2003-2004 TITLES Entire Collecion	
	Books	**Periodicals**	**Books**	**Periodicals**
Entire Institutional Library	90,000	12,403	781,021	16,234
In the following fields (included above) Engineering & Computer Science	2,352	4,100	209,456	4,836
Chemistry	320	300	5,420	327
Mathematics	336	317	1,046	337
Physics	274	163	3,264	199
Other Speciality Areas (Specify)				
Note: Periodicals are counted as books once they are bound in volumes				

TABLE VII-2.1 Library Acquisitions

	FY 2000-01	FY 2001-02	FY 20002-03	FY 2003-04	FY 2004-05
Total Library Current Funds	$1,940,378	$2,310,710	$2,313,841	2,757,391	$2,981,379
Expenditures for the Engineering Unit (Total)	$427,520	$441,434	$450,297	551,478	$462,768
Books	$14,657	$12,297	$11,751	13,787	$22,405
Periodicals-paper	$150,000	$160,000	$100,000	118,568	$80,000
Periodicals-Electronics	$88,242	$94,306	$153,316	181,988	$206,977
Other Engineering-Related Services	$174,621	$174,831	$185,230	220,590	$198,196

Table VII-2.2 Library Expenditures

The majority of library materials supporting the School of Engineering/Computer Science are held in McDermott Library, the main library of the University of Texas at Dallas. The book and periodical collections are arranged by standard Library of Congress call number. Books, including most conference proceedings, i.e., IEEE, ACM, SPIE, are located on the 4th level and are available for users to borrow.

The journal/periodical, the reference, and the standards collections are shelved on the 2nd level. These volumes must be used in the Library, but are available for reproduction within the guidelines of the copyright code. All engineering-related government publications and electronic services are available on the 2nd level in the Reference area.

The alliance for Higher Education, a regional consortium of academic and special libraries, has developed an agreement between the major libraries in the greater Dallas-Fort Worth area to make library materials available to students at cooperating institutions.

The State of Texas also maintains a consortium of academic institutions, Texshare, which provides library materials across the state.

Recognizing that the UTD Library is very young in its collection development process, telefacsimile services are excellent for the retrieval of periodical informational materials not owned. Books are also borrowed and delivered responsibly.

- UT Document Express continues to be the primary source of photocopies.

- Ariel is image software that utilizes Internet connection for document delivery world-wide.
- Texpress is an overnight delivery service of physical copies of books among the 150+ Texas University colleges.

The Eugene McDermott library has a seating capacity of 690 seats and can seat approximately 8% of the headcount student enrollment. The Engineering and Computer science reading room is unique at UTD. It has a seating capacity and desk space for 20+ students and is open 24 hours/day. The reading room has no materials acquisition budget, but receives donations of books and periodicals from faculty. It houses special collections (not cataloged) of engineering/management interest.

The McDermott Library continues to work with its professional staff and UTD faculty advisory committee to improve services and explore changes in policy. Materials are now placed on the Internet via the UTD web page electronically. Library department heads are invited to attend and participate in the regularly scheduled meetings. Continuing investigation of acquiring more engineering related journals on CD ROM will also benefit the School.

The Media Services Department was dissolved as of May 1998. The collection and classroom equipment components were formed into Audiovisual Services under control of the university libraries.

The collection is being examined and evaluated. Obsolete and damaged materials have been removed. A project is underway to purchase videos for all 16mm films. Materials are now stored in proper cases and have security devices to protect and prolong the use of the materials.

New service policies and procedures are being established to provide greater access and flexibility. Among these are no service fees for use of materials or equipment when used for classroom instruction; circulation of materials to faculty, extended hours for access to collection by students.

Funding is now being provided to faculty for purchase of audiovisual materials. A systemic approach to collection is also in place. Equipment has also been reviewed and all obsolete and irreparable items removed. New equipment meeting the current needs for classroom instruction has been purchased.

Microforms:

Microfilm	31,67	
Microfiche	1,580,217	
Microcard	2,395	
TOTAL	1,614,284	
Microform Volume Equivalents		1,755,075

Audiovisual Materials:

Videos	2,845 titles
16mm Films	283 titles
Audio CDs	83

Laserdiscs	135 titles

Standard VII-3. Systems for locating and obtaining electronic information must be available.

3. Assess the library's systems for locating and obtaining electronic information.

The McDermott Library continues to build on previous accomplishments to expand access to electronic resources provided by the University of Texas System, the Texas State Library and the Telecommunications Infrastructure Fund. On-line reference service is now available to distance education students and on-campus patrons. Teaching patrons to use technology is vital to the library's mission and the library has introduced a Bibliographic Instruction program, which is meeting that need.

The Eugene McDermott Library offers a full range of online resources to support the Erik Jonsson School of Engineering and Computer Science. The Library operates an electronic center with sixteen workstations designed to support Internet research activities and to provide access to a collection of compact disc materials.

At present, the Library subscribes to over 200 Internet-based resources, many of which include the full text of periodical or newspaper articles and complete statistical/numerical data from major publishers such as the U.S. federal government, the United Nations, Moody's Financial Services, Commerce Clearing House, and Disclosure Inc. The Internet database collection is available off-campus to UTD students and faculty except when the information provider updates their systems. This mode of access supports all distance earning activities. Most compact disc products must be used in the Library, although the Department actively migrates products to the Internet as needed.

The Library also offers database search capabilities of online resources available from DIALOG and WESTLAW. Fees are charged on a cost recovery basis.

Describe the process by which acquisitions of engineering-related materials are made.

Library materials for the School of Engineering and Computer Science are acquired through a campus-wide process. An annual allocation is made for the acquisition of books, reference resources, audiovisual/videos, and the filling of gaps as necessary in the journal collection. "On Approval" books and notices are made available to the school automatically through a well-detailed profile representing the faculty research and course instruction programs. The engineering librarian is responsible for the monitoring of this program and orders materials through a wide variety of review materials.

Standard VII-4. Classrooms must be adequately equipped for the courses taught.

B. Classroom Equipment. Describe the equipment typically available in classrooms where you teach your courses. Assess its adequacy for the purpose.

Almost all Computer Science classes are taught in the new South wing to the Engineering and Computer Science Complex (opened in August 2002). This building includes 10 state of the art classrooms with seating capacities of 180 (1), 120 (3), 80 (4), and 70 (2); the four larger classrooms have amphitheater-style seating. The main auditorium was used for large classes but is now reserved for special events. The classrooms have a podium that includes a PC with network access, a document reader (both hardcopy and transparency), and connections for laptops. The projection system can display input from the document reader, PC, laptop offering instructors a full range of options. Each student has access to a power outlet and a microphone; the 4 larger classrooms have TV monitors that mirror the main projection screen. Classrooms are equipped with cameras enabling taping of lectures.

The limited white-board space in the larger classrooms (due to the large permanent projection screens) has posed a challenge to some faculty but we have worked around it through scheduling and use of the document reader.

Standard VII-5. Faculty offices must be adequate to enable faculty members to meet their responsibilities to students and for their professional needs.

C. Discuss and assess the adequacy of faculty offices to enable faculty to meet their responsibilities to students and for their professional needs.

Computer Science faculty offices are in the South Wing of the Engineering and Computer Science Complex since it opened in August 2002. The typical office is about 400 square feet and includes a desk, computer table, bookcases, filing cabinets, whiteboard, and two network outlets. All offices are on the perimeter of the building. Office facilities are actually a recruiting tool as they compare favorably to most other Universities.

A. General Information:

Name of Institution: The University of Texas at Dallas Date: 6/27/05

Address: 2701 Floyd Road, Richardson, TX 75080
(Street) (City) (State) (Zip)
URL: http://www.utdallas.edu

Name and Title of Chief Executive Officer of Campus (President, Chancellor, etc.)
Dr. David E. Daniel, President
(Name) (Position)

B. Type of Control (Check more than one, if necessary)

______Private, non-profit	______Municipal
______Private, other	__x__State
______Federal	

Affiliation, if private: ______________________

If above classifications do not properly apply to the institution, please describe its type of control.

C. Regional or Institutional Accreditation. Name the organizations by which the institution is now accredited, give dates of most recent accreditation. Attach a copy of the most recent accreditation action by any organization accrediting the institution or any of its computer-related programs.

UT-Dallas is accredited by the Southern Association of Colleges and Schools (initial accreditation in 1972; most recent on December 8, 1998).

D. Enrollment

Total enrollment for the entire institution (FTE) 13,316 (5,381 part-time)

Total faculty for the entire institution (FTE) 472 full-time; 225 part-time

E. Funding Process. Describe the process for allocating institutional funds to the computer science program.

Insitutional funds reside with the Erik Jonsson School administration, which decides the proportion of funds to be allocated to the two departments in the School. Appropriations to the departments are generally made in accordance to their needs (which generally are in proportion of their respective sizes). For instance, the funding for TA positions is done in proportion to the total number of enrolled student credit hours in a particular semester in each school. The need, however, may not be dependent on the size of the department. For instance, equipment expenditure incurred by the EE department is much larger due to the expensive nature of the equipment involved. In contrast, equipment in CS consists mostly of computers, which are becoming quite inexpensive. Thus, the equipment budget of the EE department is significantly larger than that of the Computer Science Department.

F. Promotion and Faculty Tenure. Summarize the promotion and tenure system and the system for merit salary adjustments. (Give an overview of actual practice; do not reproduce an entire section from the faculty handbook.)

Tenure reviews are conducted during the 6th year of service (a faculty may request an earlier review). Promotion to Associate Professor and tenure are usually awarded together. The promotion and tenure review involves an Ad Hoc Committee, solicitation of independent external reviews, votes by the above-rank faculty in the Department, recommendation by the faculty, Dean, and the Committee on Qualifications of Academic Personnel. Contributions in the areas of research, teaching, service are evaluated with adequate performance expected in all three areas and excellent performance expected in research or teaching for a successful review.

Promotion to full professor does not follow a prescribed time-table but is not usual for less than 5 years of service after tenure. The process is similar to the tenure review.

Annual evaluations of all faculty are performed. The Department's Personnel Review Committee makes the initial evaluation; the Dean, with input from the Department Head makes the final evaluation and recommends merit salary increases to the Provost.

G. Retirement and Benefits. Summarize the retirement program and other faculty benefits.

Faculty members are paid competitive salaries (cf. Taulbee Survey on Faculty Salaries conducted by the Computing Research Association every year). Faculty members are provided standard benefits such as health insurance, dental coverage, vision care, long term disability insurance, accidental death and dismemberment insurance, life insurance, and pension. Faculty members have the option of either joining the Teacher Retirement System (TRS) or the Optional Retirement Plan (ORP, essentially a 403(b) plan). UTD contributes 6% in both schemes. 6.4% is deducted from the faculty members pay for TRS and 6.65% for ORP. UT Dallas also pays the employer portion of social security/FICA tax.

Ref: http://www.utdallas.edu/utdgeneral/business/hr/benefits_at_ut__dallas.htm

Appendix II. General Information on the Unit Responsible for the Computer Science Program

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

A. Type of unit

1. Name of computer science program unit: Department of Computer Science

URL http://www.utdallas.edu/dept/cs/

2. If the computer science administrative program unit is not a department reporting to an administrative officer (e.g., Dean of College of Arts and Sciences) who in turn reports to president, provost, or equivalent executive officer, describe the unit.

B. Administrative Head of Computer Science Program Unit

D.T. Huynh	Professor and Department Head
(Name)	(Title)

C. Organization Chart. Attach an organization chart showing how the unit fits into the administrative structure of the institution.

.

Table A-II.1. UT Dallas Administrative Structure

The dean of the Erik Jonsson School reports to the Provost; the Provost reports to the President.

President's Cabinet ═══ President's Senior Staff ....

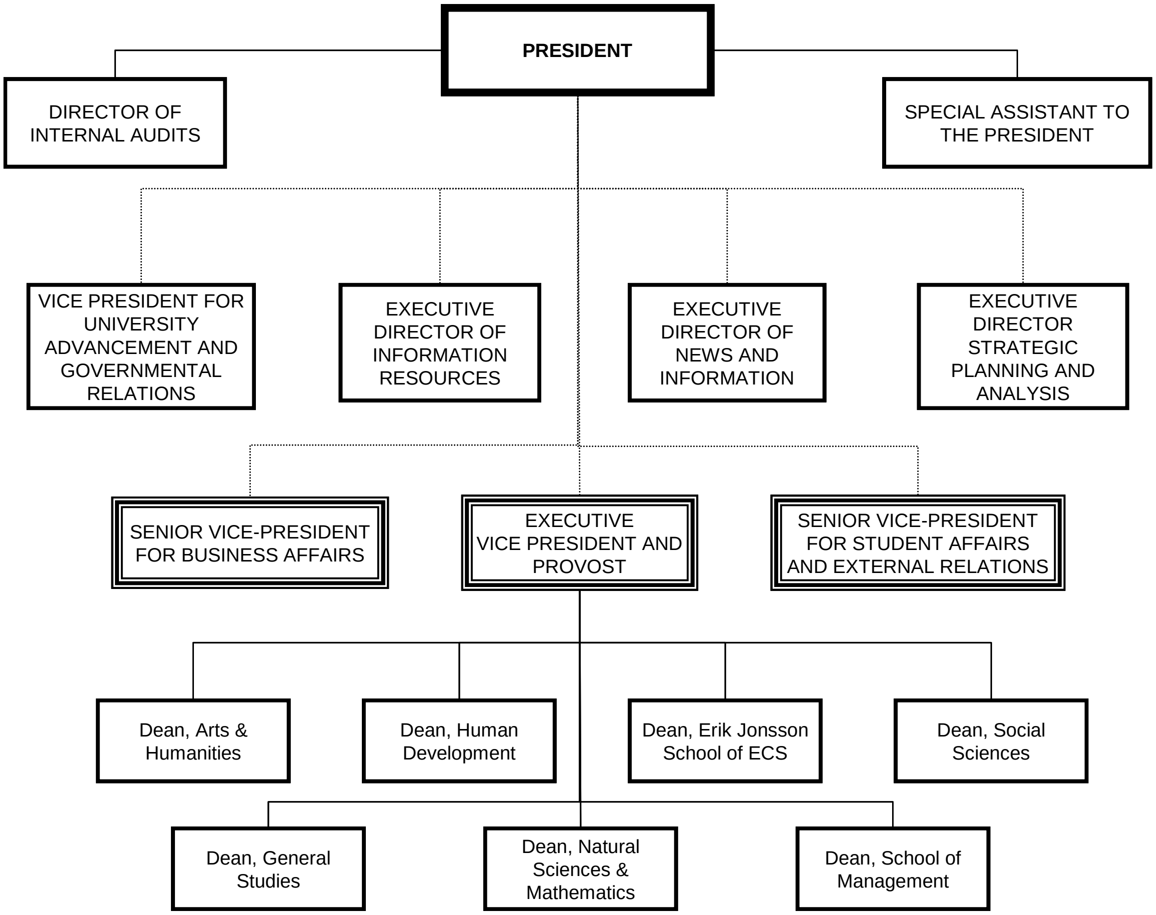
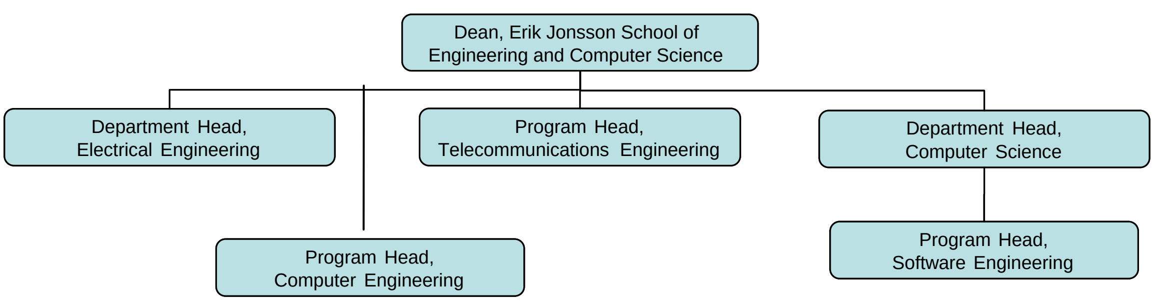


Table A-II.2. Engineering Organizational Chart



Academic Subdivisions and Department/Program Heads

- Erik Jonsson School of Engineering and Computer Science, Dr. Robert Helms, Dean
- Department of Computer Science, Dr. D.T. Huynh
- Department of Electrical Engineering, Dr. John Hansen, Department Head and Professor in Electrical Engineering
- Telecommunications Engineering Program, Dr. Subbarayan Venkatesan, Program Head and Professor in Electrical Engineering
- Computer Engineering, Dr. Dinesh Bhatia, Program Head and Professor in Electrical Engineering
- Software Engineering, Dr. Simeon Ntafos, Program Head, Associate Dean for Undergraduate Education, and Professor of Computer Science.

Several research centers and labs have been established in both the Department of Electrical Engineering and the Department of Computer Science. These centers have acquired large scale projects through cooperation with industry, government and other academia institutions.

Formal Research Centers in the Erik Jonsson School

- Photonic Technology and Engineering Center (PhoTEC), directed by Dr. Cy Cantrell, Professor of Electrical Engineering.
- Center for Systems, Communications and Signal Processing (CSCSP), directed by Dr. Bob Hunt, Professor of Electrical Engineering.
- Center for Integrated Circuits and Systems (CICS), directed by Dr. Poras Balsara, Professor in Electrical Engineering.
- Embedded Software Center (ECS), directed by Dr. Farokh Bastani, Professor in Computer Science
- Human Language Technology Research Institute (HLTRI), directed by Dr. Sanda Harabagiu, Associate Professor in Computer Science.
- Security Analysis and Information Assurance Laboratory (SAIAL) and Cyber Security Research Center, directed by Dr. Bhavani Thuraisingham, Professor of Computer Science.
- Digital Forensic and Emergency Preparedness Institute (DFEPI), directed by Dr. E. Douglas Harris, Associate Dean, Erik Jonsson School of Engineering and Computer Science.

Research Labs in the Department of Computer Science

1. Distributed Systems Lab
2. Applied Logic, Programming-Languages and Systems Lab (ALPS)
3. Software and Information Visualization Lab

4. DSP and Communications Lab
5. Wireless information systems Lab
6. Multimedia Distance Learning Lab
7. Parallel Computation Lab
8. Artificial Intelligence Lab
9. CAD and Visualization Lab
10. UTD Database Laboratory
11. Telecommunications Lab
12. Computer Vision and Multimedia Systems Lab
13. Telecommunications and SE Lab
14. Resource Allocation and Scheduling Lab
15. Laboratory of Advanced Computer and Network Architectures
16. Advanced Networking and Dependent Systems Laboratory
17. Multimedia Systems and Networking Lab
18. Software Technology Advanced Research
19. Compiler and Architecture Research Lab
20. NET Lab - Scalable Network Engineering Techniques Laboratory
21. Visual Computing
22. Formal Method Lab
23. Software Architecture Lab
24. Advanced Network Research Lab
25. Advanced Computation Lab
26. Requirements Engineering Lab
27. Distributed and Internet Computing Lab

Research Labs in the Department of Electrical Engineering

1. Cochlear Implant Laboratory
2. Digital Signal Processing Laboratory
3. Embedded and Adaptive Computing Group
4. Multimedia Communications Laboratory
5. Optical Communications Laboratory
6. Optical Networking Advanced Research (OpNeAR)
7. Plasma Applications Laboratory

8. Speech Processing Laboratory
9. MicroNano Devices and Systems Laboratory (MiNDS)
10. Micro-device Research Laboratory (MdRL)

D. Research Organizations. Describe any research organizations, institutes or other related facilities that are part of the unit responsible for the computer science program or that closely affect its operation.

The Department is within the Erik Jonsson School of Engineering and Computer Science. Research Centers, Institutes, and Laboratories are listed in the previous section.

E. Computer-Related Undergradate Degree Programs. List all undergraduate computer-related degree programs offered by the institution, beginning with the program(s) being evaluated.

Program Title	Years Required	Degree Awarded	Administrative Unit	If accredited, by whom
Computer Science	4	B.S.C.S.	CS Dept	
Software Engineering	4	B.S.S.E.	CS Dept	
Electrical Engineering	4	B.S.E.E.	EE Dept	ABET EAC
Telecommunications Engineering	4	B.S.T.E.	EE Dept	ABET EAC

Are these programs adequately differentiated in all university information? Explain how.

Computer Science and Software Engineering

The Computer Science Department offers the B.S. degree in Computer Science and the B.S. degree in Software Engineering. Both are based on a solid foundation of mathematics, including calculus, linear algebra, and discrete mathematics. These programs of study are designed to offer students opportunities to prepare for an industrial, business, or governmental career in a rapidly changing profession and to prepare for graduate study in a field in which further education is strongly recommended. The two programs have the same basis in core computer science, including the analysis of algorithms and data structures, modern programming methodologies, and the study of operating systems. The Computer Science program continues with courses in advanced data structures, programming languages, telecommunications networks, and automata theory, while the Software Engineering program includes courses in requirements engineering, software validation and testing, and software architecture, culminating in a challenging project course in which students must demonstrate use of software engineering techniques. Both programs offer a rich choice of elective studies, including courses in artificial intelligence, computer networks, computer graphics, databases, and compiler design.

Electrical Engineering and Telecommunications Engineering

The Electrical Engineering Department offers two engineering programs: Electrical Engineering and Telecommunications Engineering. The Electrical Engineering program offers students an opportunity to acquire a solid foundation in the broad areas of electrical engineering and emphasizes advanced study in digital systems, telecommunications, and microelectronics. The Electrical Engineering program offers students a solid educational foundation in the areas of electrical networks, electronics, electromagnetics, computers, digital systems, and communications and is accredited by the Engineering Accreditation Commission of the Accreditation Board for Engineering and Technology (ABET). Mastery of these areas provides students with the ability to adapt and maintain leadership roles in their post-baccalaureate pursuits through the application of fundamental principles to a rapidly changing and growing discipline.

Students in the Electrical Engineering program take either the general program or specialize in microelectronics or telecommunications, and can take advanced courses in computer hardware and software; the analysis and design of analog and digital communication systems; analog and digital signal processing; the analysis, design, and fabrication of microelectronic components and systems; and guided and unguided wave propagation. A broad choice of electives (within and external to electrical engineering) allows students to broaden their education as well as develop expertise in areas of particular interest. In keeping with the role of a professional, students are expected to develop communication skills and an awareness of the relationship between technology and society.

The Telecommunications Engineering program is interdisciplinary. Telecommunications Engineering requires a blend of knowledge from the areas of Electrical Engineering, Computer Science, and Economics/Policy.

The Electrical and Telecommunications Engineering programs are based on a solid foundation of science and mathematics coursework. Students in these programs are given an opportunity to learn to extend their abilities to analyze and solve complex problems and to design new uses of technology to serve today’s society. The engineering programs provide an integrated educational experience directed toward the development of the ability to apply pertinent knowledge to the identification and solution of practical problems in electrical and telecommunications engineering. These programs ensure that the design experience, which includes both analytical and experimental studies, is integrated throughout the curriculum in a sequential development leading to advanced work. Design problems are frequently assigned in both lecture and laboratory courses. Each student is required to complete a major design project during the senior year. In addition, established cooperative education programs with area industry further supplement design experiences.

Appendix III. Finances

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program and clearly delineate which program is being described.

A. Finances Related to the Computer Science Program(s).

1. For the computer science program, indicate below the funds expended during the fiscal year immediately preceding the visit [1].

	Institutional Funds	Non-recurring or Outside Funds
Administrative Salaries	$215,000	
Faculty Salaries	$4,626,000	
Non-teaching Professionals' Salaries[2]	$100,000	$535,000
Support Personnel Salaries & Wages Secretarial	$ 210,000	
Technician	$ 153,400	
Other (specify)		
Graduate Students	$ 678,436	
Operating Expenditures (excluding research operations and travel)	$ 100,000	
Capital Equipment Expenditure: (including value of allocated time for teaching and research):		
Teaching		
Research		
Computer Expenditures: (total, including value of allocated computer time for teaching and research) Hardware	$205,000	
Software	$ 30,000	
Allocated time 100%		
Travel Expenditures (non-research funds)	$ 20,000	$109,221
Scholarship Awards (if administered by the Computer Science Program Unit)	$40,000	$130,000
Library (if administered by Computer Science Program Unit)		
Research (if separately budgeted)		$1,426,849
Other (specify)		
Total	$6,377,836	$ 2,201,070

[1] It is understood that some of the data may have to be estimated to cover the entire fiscal year. In such case, unless the differences are insignificant, an updated report should be provided for the evaluation team at the time of the visit.

[2] Non-teaching professionals would include research professors, faculty members on paid sabbatical leave, post-doctoral research associates, and other degreed professionals.

2. Report funds for the fiscal year immediately preceding year of visit, broken down according to source.

	AMOUNT
Institutional funds (recurring)	$6,377,836
Gifts and non-research grants	
Research contracts and grants	$2,201,070
Other (explain)	
Total	$ 8,578,906

B. Operating and Computing Expenditures for the Five Fiscal Years Immediately Preceding that Reported in III A.

1. Operating expenses for the computer science program unit.

Fiscal Year					
Institutional Funds					
Outside Funds					

See the tables in C below

.

2. Computer hardware/software capital expenditures (excluding equipment used primarily for research) for the computer science program unit.

Fiscal Year					
Institutional Funds					
Outside Funds					

See the tables in C below.

C. Additional Funding. If additional funds, other than those listed in Table A.1 above, are available to faculty to support scholarly activities such as travel to technical meetings, e.g., consulting support, give the number of faculty for whom this type of support is appropriate and an estimate of the amount of support available.

The expenditure tables for the whole school, the Computer Science department, the Electrical Engineering Department are shown in the next 3 tables. The fourth table shows expenditures that are not directly attributable to either CS or EE. Accounting and finance management is done largely at the School level, as a result it is rather hard to separate CS and EE expenditures, especially given that several research programs as well as degree programs span the two department. The tables have been generated with the help of expenditure codes used to classify expenditure in all accounts. So this data may appear a little inconsistent with financial data reported earlier. However, that is because expenditures reported above may be accounted for under different expenditure codes. For example, money spent on actual equipment will be accounted for in the equipment category as well as institutional funds category below. The tables below are included to show that the Computer Science program and the Engineering and Computer Science school are more than adequately funded. Note that 2004-2005 expenditures are low because those accounts have not been closed yet so all expenses have not been posted yet.

	ECS Combined			
Fiscal Year	**2001-2002**	**2002-2003**	**2003-2004**	**2004-2005**
Expenditure Category				
Operations[1](not including staff)	4,184,358	4,125,844	4,816,900	5,638,921
Travel[2]	393,079	304,920	488,916	342,814
Equipment[3]	1,077,180	1,791,304	5,097,064	4,875,894
Institutional Funds	1,400,394	1,570,515	5,360,284	2,972,589
Grants and Gifts[4]	4,254,223	4,656,553	5,042,596	7,885,040
Graduate Teaching Assistants	1,554,174	1,718,671	1,729,682	1,343,331
Part-time Assistance[5] (other than teaching)	688,038	652,459	1,381,157	1,185,434

Table A-III.1 Support Expenditures *ECS School Total Expenses*

	Computer Science			
Fiscal Year	**2001-2002**	**2002-2003**	**2003-2004**	**2004-2005**
Expenditure Category				
Operations[1](not including staff)	527,118	681,156	938,951	1,389,496
Travel[2]	70,415	75,537	125,730	129,221
Equipment[3]	19,703	103,237	46,577	27,003
Institutional Funds	259,931	277,442	683,832	118,871
Grants and Gifts[4 *]	357,305	587,488	427,426	1,426,849
Graduate Teaching Assistants	1,070,858	993,693	1,058,647	678,436
Part-time Assistance[5] (other than teaching)	186,110	217,169	337,098	183,842

Table A-III.2 Support Expenditures *Direct Computer Science Expenses*

	Electrical Engineering			
Fiscal Year	**2001-2002**	**2002-2003**	**2003-2004**	**2004-2005**
Expenditure Category				

Operations[1](not including staff)	955,755	1,444,661	1,698,227	1,163,405
Travel[2]	130,921	114,217	144,703	82,043
Equipment[3]	289,600	262,665	179,764	54,907
Institutional Funds	292,905	297,960	380,284	231,284
Grants and Gifts[4]	1,083,371	1,523,583	1,642,410	1,069,071
Graduate Teaching Assistants	483,316	724,978	671,035	586,349
Part-time Assistance[5] (other than teaching)	108,860	130,162	431,796	301,197

Table A-III.3 Support Expenditures *Direct Electrical Engineering Expenses*

	Undistributed Central School			
Fiscal Year	**2001-2002**	**2002-2003**	**2003-2004**	**2004-2005**
Expenditure Category				
Operations[1](not including staff)	2,701,485	2,000,027	2,179,722	3,086,020
Travel[2]	191,743	115,166	218,483	131,550
Equipment[3]	767,877	1,425,402	4,870,723	4,793,984
Institutional Funds	847,558	995,113	4,296,168	2,622,434
Grants and Gifts[4]	2,813,547	2,545,482	2,972,760	5,389,120
Graduate Teaching Assistants	0	0	0	78,546
Part-time Assistance[5] (other than teaching)	393,068	305,128	612,263	700,395

Table A-III.4 Support Expenditures *Undistributed Central School Accounts*

5 Research Assistants

Note: All 2005 figures are as of March 31, 2005. Part-time Assistance represent Research Assistants.
Equipment includes all capital expenses.

<u>Data Collection Timeframes and Definitions:</u>
Fiscal Year 1998-1999 (Fall 1998 Spring 1999 & Summer 1999)
Fiscal Year 1999-2000 (Fall 1999 Spring 2000 & Summer 2000)
Fiscal Year 2000-2001: (Fall 2000, Spring 2001, & Summer 2001)
Fiscal Year 2001-2002: (Fall 2001, Spring 2002, & Summer 2002)
Fiscal Year 2002-2003: (Fall 2002, Spring 2003, & Summer 2003)
Fiscal Year 2003-2004: (Fall 2003, Spring 2004, & Summer 2004)

* Instructor: Non-T/Tenure Track full-time teaching faculty,
** Faculty members: All T/Tenure Track and full-time teaching faculty, employed full-time since the Fall 2000 semester.

Notes:

1. General operating expenses to be included here
2. Institutionally sponsored, excluding special program grants.
3. Major equipment, excluding equipment primarily used for research.

Note that the expenditures under "Equipment" should total expenditures for Equipment.
If they don’t, please explain.

4. Including special (not part of institution's annual approbation) non-recurring equipment purchase programs.
5. Do not include graduate teaching and research assistant or permanent part-time personnel.

Appendix IV. Computer Science Program Personneland Policies Towards Consulting, Professional Development, and Recruiting.

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

A. Term of appointment of administrative head.

9 month______ 12 Month__X___ Other (specify) ____________________

B. Number of personnel associated with program.

	Full-time Number	Part Time Number	Part Time FTE	Total FTE
Faculty	56	2	1	57
Non-teaching Professionals	6			6
Administrative				
Computer Lab Personnel:				
Professionals	3			3
Technicians				
Secretarial, Accounting, etc.	8			8
Graduate Teaching Assistants		65		32.5
Graduate Research Assistants		60		30
Graduate Students	556 (total)			
Undergraduate Students	1002 (total)			

C. Policies. Provide a brief description to give an overview.

1. Describe policy toward private consulting work, sponsored research projects, and extra compensation.

   Consulting is allowed (up to one day a week). Obtaining external funding (industry, government) is highly encouraged.

2. State the standard teaching, administrative, research, and other loads on the faculty, in general terms.

   The standard teaching load for tenure track faculty is two organized classes each semester (including at least one undergraduate class per year). However, tenure track faculty have been able to reduce their teaching load to 3 or even 2 courses a year using credits for supervision of PhD students. Faculty members with significant external funding may buy-out of up to one course per year. The teaching load for Senior Lecturers is 4 organized classes each semester. (Summer is not included in the discussion above).

Other than the Department Head and Associate Heads (3 total), faculty do not have administrative responsibilities. In addition to teaching, tenure-track faculty are expected to do research (50%) and service to the department, university, community (10%).

3. Describe policies and procedures for recruiting faculty for the computer science program. Describe any barriers to hiring the appropriate faculty.

Faculty recruiting starts with meetings of senior faculty and the Department Head with the Dean in summer to identify focus areas for the next academic year's search. Search committees are appointed in early Fall and advertisements are placed in various outlets (Chronicle of Higher Education, Communications of the ACM, IEEE Computer, Computing Research Association Newsletter); in addition, applications are solicited from minority institutions. The search committee reviews and evaluates applications, requests reference letters, and identifies candidates to invite for interviews (usually with input from the faculty). A typical interview visit includes a seminar and individual meetings with faculty. Feedback is collected from the faculty, evaluated, and the search committee recommends offers to the faculty; in turn, the faculty recommends offers to the Dean. Final offers are made by the Provost after the file is reviewed by a University-wide Committee on Qualifications of Academic Personnel (which adds its own recommendation**).**

In general, the Department of Computer Science has been very successful in recruiting over the past 5 years. Not only were we able to double the size of the tenure-track faculty but the quality of recruited faculty also increased. The main barrier to attracting even higher caliber faculty is the candidate's preference to accept offers from higher ranked programs**.**

Appendix V.
Computer Science Program Enrollment and Degree Data

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

Give below enrollment figures for the first term of the current and five previous academic years and the number of undergraduate and graduate degrees conferred. (The current year is the year in which this report is being prepared.) List data beginning with the most recent year first. If part-time students are involved, give the number as FTE/actual number, e.g., 10/40.

Engineering and Computer Science School as a whole

Enrollment and Degrees for the Erik Jonsson School (as a whole)

Year	AY	FT/ PT	Enrollment Year					Total UG	Total Grad	Degrees Conferred			
			1st	2nd	3rd	4th	5th*			BS	MS	PhD	Other
04	2005	FT	417	207	331	369	6	1330	599	397	433	24	
		PT	25	53	145	254	33	510	415				
03F	2004	FT	422	218	372	392	0	1404	727	330	441	14	
		PT	33	67	185	294	8	625	493				
02F	2003	FT	472	231	370	378	17	1468	802	372	357	9	
		PT	38	69	184	291	67	649	493				
01F	2002	FT	518	222	345	358	22	1465	786	321	318	15	
		PT	64	77	202	310	71	724	381				
00F	2001	FT	378	189	301	325	16	1209	549	327	307	10	
		PT	58	62	191	276	69	656	380				
99F	2000	FT	323	128	283	319	19	1072	352	241	311	9	
		PT	40	77	173	276	42	608	366				

* The fifth enrollment year is the number of post-baccalaureate students taking undergraduate courses.

Unit offering Computer Science Program(s)—give total enrollment even if not all students are in the program for which accreditation is requested.

Degrees for Computer Science (incl. Software Engg major)

Year	AY	FT/PT	Enrollment Year 1st	2nd	3rd	4th	5th*	Total UG	Total Grad	Degrees Conferred BS	MS	PhD	Other
04	2005	FT	255	121	173	206	4	758	324	272	297	6	
		PT	11	21	62	146	14	244	237				
03F	2004	FT	259	123	221	246	0	849	410	238	344	10	
		PT	22	33	89	176	1	345	314				
02F	2003	FT	271	129	226	275	12	883	546	267	282	3	
		PT	26	34	112	182	48	402	320				
01F	2002	FT	311	138	219	249	13	930	591	214	351	10	
		PT	42	55	131	183	48	459	255				
00F	2001	FT	253	127	201	201	11	793	429	246	214	1	
		PT	40	41	123	166	48	418	223				
99F	2000	FT	202	90	172	215	14	693	269	155	236	3	
		PT	27	56	113	181	32	409	185				

If the unit offering the computer science program(s) offers more than one degree, please complete an additional table for each program for which accreditation is requested:

Degrees for Computer Science

Year	AY	FT/PT	Enrollment Year 1st	2nd	3rd	4th	5th*	Total UG	Total Grad	Degrees Conferred BS	MS	PhD	Other
04	2005	FT	204	96	151	194	3	648	274	266	218	6	
		PT	10	19	55	123	13	220	182				
03F	2004	FT	197	108	208	237	0	750	342	234	265	10	
		PT	17	30	82	169	23	321	239				
02F	2003	FT	243	120	221	242	9	835	460	267	207	3	
		PT	24	31	105	177	43	380	248				
01F	2002	FT	309	137	219	248	13	926	537	214	160	10	
		PT	42	53	126	183	45	449	222				
00F	2001	FT	253	127	201	201	11	793	386	246	178	1	
		PT	40	41	123	166	46	416	199				
99F	2000	FT	202	90	172	215	14	693	261	155	236	3	
		PT	27	56	113	181	32	409	183				

Appendix VI. Admission Requirements

If you are having more than one program evaluated, particularly if the programs are on separate campuses, the answers to these questions may vary from one program to another. If this is the case, please use separate copies of this section for each program, and clearly delineate which program is being described.

A. Admission of students

1. Describe the criteria and procedures used for admitting students to the computer science program(s).

U.T. Dallas accepts applications for admission from freshmen and transfer students at all levels for the fall, spring, and summer semesters. The Admissions and Records Office is located in the lower level of the Eugene McDermott Library.

Applicants are invited to contact the Office of Enrollment Services, located in the University Conference Center, for counseling and specific information about degree program prerequisites that may enable them to qualify for admission. Early contact ahead of the college freshman year will enable applicants to take advantage of admissions counseling in order to make a wise selection of courses prior to enrollment at U.T. Dallas

The relevant web site for this information is:
http://ospa.utdallas.edu/cds/2000/frosh.html
http://www.utdallas.edu/dept/ugraddean/freshman.shtml

Entering Freshmen
The curriculum and the expectations of student performance at The University of Texas at Dallas assume that entering freshman students have successfully completed a full college-track high-school curriculum and have demonstrated strong general verbal/quantitative aptitudes as measured on national standardized test.

Automatic Admissions
In accordance with Chapter 51 of the Texas Education Code, students are automatically admitted to the University as first-time freshmen if they graduate in the top 10% of their class from an accredited Texas high school. Applicants must have graduated from high school during one of the two school years preceding the academic year for which they seek admission and be considered first time freshman. Applicants admitted because they are in the top 10% of their high school class may be required to complete additional preparatory work before enrolling in the University. They may also be required to remove any deficiencies in their high school coursework before graduating from the University.

Reviewed Admissions
Applications from all students not graduating from Texas high schools in the top 10% of their class, as specified above, will be reviewed. Applicants must have graduated from an accredited high school or satisfied equal requirements, and should have completed the high school unit

requirements listed below (See item 9). Admission decisions are based on the applicant's composite achievement profile, including:

- high school class rank;
- strength of academic preparation including the number of courses taken and their difficulty (honors, AP, IB, etc);
- SAT-I or ACT scores;
- record of achievements/honors/awards;
- special accomplishments/work/service both in and out of school;
- essays (view topics);
- special circumstances that put academic achievements in context;
- recommendations (not required);
- successful completion of a high school curriculum that includes:

    - four units of Language Arts, including at least one unit of writing skills;
    - two units of a single foreign language (three units recommended);
    - three and one-half units of Mathematics beginning with Algebra I or higher and including a course dealing with trigonometry, such as pre-calculus (four units recommended);
    - three units of laboratory science, not including Physical Science;
    - three units of Social Sciences, not including work-study (four units recommended);
    - one-half unit of Fine Arts (one unit recommended);
    - one and one-half units of General Education Electives (two and one-half units recommended);

The University also recommends one unit of Computer Science, one-half unit of Health, and one and one-half units of Physical Education. For Texas residents, consideration may be given to socioeconomic and geographic information.

The review process gives primary consideration to the applicant's scores on standardized tests and high school record although no specific class rank, test score, or other qualification by itself assures admission except as described above under Automatic Admissions. The decision for each applicant will be to approve admission or to deny admission.

The achievement levels of students admitted to UT-Dallas are illustrated by the following statistical profile of the entering freshman class of Fall 1999.

61% of students were in the top 25% of their high school graduating class;
29% were in the top 10% of their class;
50% of students scored between 1090 and 1300 on the SAT-I;
26% scored 1300 or higher (the 1998 national average SAT-I is 1017)

Reference:General Admission Info http://www.utdallas.edu/dept/ugraddean/admissions.shtml

Math Placement:
All students who intend to take pre-calculus, applied calculus, or calculus I must qualify for admission to the class based on their performance on the SAT II math tests or other credit by examination. On the second day of Freshman Orientation, the Learning Resource Center will administer both the SAT II IC and IIC mathematics tests. Students wishing to enter pre-calculus must attain a score of 460 on the SAT II IC test. To enter applied calculus, a student must score at least a 480 on the SAT II IC. For calculus I admission, students must score at least 560 on the SAT II IC tests or a 530 on the SAT IIC test.

Reference: Math Placement http://www.utdallas.edu/dept/ugraddean/mathplacemt.shtml

2. Describe procedures, including the evaluation of transfer credits, for students admitted to the program as transfer students

   a. from within the institution

Transfer from within the institution occurs when a student changes major, either from Undeclared status or from another academic program. In order to change major to Computer Science, a student must meet the requirements set for all new applicants to the major.

   b. from another institution

UT Dallas welcomes applications from prospective students who have begun their college work at other institutions of higher education. To be admitted the applicant must be in good standing at the institution(s) previously attended. The rule is that transfer credit is allowed for courses from accredited institutions that have content that is judged to be equivalent to that offered at UT Dallas. Catalog descriptions, course outlines, books and other documentation are used to check equivalence. Each case is considered individually, and checked by the Associate Dean for Undergraduate Education (Dr. Simeon Ntafos). The Office of the Registrar also evaluates every request for transfer of upper level credit. The university accepts for transfer credit only academic post-secondary course work completed with a grade of C or better at accredited institutions of higher education. UT Dallas does not offer credit for nonacademic course work, such as vocation, developmental or remedial studies, nor grant credit for prior experiential learning. Course work that is accepted for transfer credit is applicable toward satisfying requirements for a specific UTD major according to the same criteria as those used for equivalent UTD courses. Prospective transfer students for Dallas-area community colleges should refer to the UTD 2+2 Transfer Guide, available at community college counseling offices and at the UTD Office of Enrollment Services, in order to inform themselves about curricula appropriate to the various UTD majors.

As soon as an application for admission, including transcripts and any required test scores, has been received, the Office of Admissions and Records will evaluate the student's record to determine which credits earned at another college or university will transfer to UTD. Each student who is admitted to UTD will receive a copy of this evaluation and an outline of the degree plan for the program to which the student is admitted.

A copy of the evaluation will also be sent to the student's advising office. An undergraduate advisor, in consultation with the Associate Dean for Undergraduate Education, will determine how the transfer credits apply to UTD graduation requirements. The application of transfer credit to degree plans must be completed within the first semester of enrollment. The faculty, acting through the Associate Dean for Undergraduate Education, have the ultimate responsibility for applying transfer credit to the requirements for specific academic degrees. Students are urged to contact their advising office upon receipt of the letter informing them of their admission to UTD. See, also, the section on the Texas Academic Skills Program.

Freshmen and Sophomores

Applicants to UTD who have previously taken courses at one or more other accredited institutions of higher education and who are classified as freshmen or sophomores will be reviewed for admission using the same criteria for first-time freshmen. Freshman applicants must have a cumulative GPA of at least 3.00, on a 4.00 scale, for all post-secondary academic course work. Sophomore applicants must have a cumulative GPA of at least 2.50, on a 4.00 scale, for all post-secondary academic course work.

Juniors and Seniors

Automatic Admissions

Applicants to UTD who have previously taken courses at one or more other accredited institutions of higher education and who are classified as juniors or seniors are admitted automatically if their cumulative GPA for post-secondary academic course work is 2.50 or better, on a scale of 4.00.

Reviewed Admissions

Applications that do not qualify for automatic admission will be reviewed at the discretion of the Associate Deans for Undergraduate Education of the college housing the applicant’s major. Associate Deans for Undergraduate Education will pay particular attention to the academic content and grades of the applicant’s college-level work.

Students admitted on probation must earn a GPA of as least 2.20 for the first semester of enrollment. Failure to meet this condition will result in suspension. Students admitted on probation by the college master who are subsequently suspended from the university may be readmitted only by the college master.

Regardless of the number of lower-division hours which a student transfers to the university, applicants seeking admission to UTD should be aware that they will need at least 51 upper-division hours to graduate.

Program Prerequisites

In addition, for admission to a degree program, the student should have completed all lower-division prerequisites as determined by the program. The student should consult the program listings in the Undergraduate Catalog for these requirements. Excellent resources for local community college students are the 2+2 Guides, which detail both program prerequisites and General Education Core courses by the community college course numbers. These guides are available at the UT Dallas Office of Enrollment Services, the Admissions and Records Office, or from local community college counselors.

3. Explain the policy of the institution in admitting students with conditions and state how the conditions must be made up.

See 2 above.

4. Describe the general policy and methods of the unit offering computer science program(s) in regard to admission with advanced standing.

See 2 above.

5. Describe any special admission requirements for entry into the "upper division" in the computer science program(s).

See 2 above.

APPENDIX VII
GUIDELINES FOR IN-CLASS ASSESSMENTS

Prepared by: Simeon Ntafos **Last Update**: 11/17/04

1. Educational objectives for required CS, SE classes have been set up and in-class assessment forms filled out since Summer 2002. Summaries for the period U02-S04 were prepared by the Class coordinators in early October 2004 and reviewed by the UG Curriculum Committee which prepared a report. Dr. S.Q. Zheng presented at the Faculty meeting on 10/29/04).

   Educational objectives for almost all CS, SE classes are done, including electives, and classes taught by Math, Cognitive Sciences. We are missing one active class, one existing class, and several new electives that have not been taughtyet. The available educational objectives have been posted in the Department's webpage (under "Accreditation") where all stakeholders have access to them. The educational objectives of the remaining CS classes are needed.

2. There are many gaps in the data collected so far. We need 100% response rate for Fall 2004 and in the future. In the past, some forms have been turned inmonths after the class was over. These assessment forms have to be completed and turned in by the time grades are due for the semester. Class coordinators will review them to make adjustments by the time the next semester starts.

   Starting with Fall 2004, in-class assessment will include all CS, SE classes as well as other classes that are critical to the ABET effort (e.g., ISSS 3360).

3. The class learning objectives and the mapping to the ABET objectives (a-k) are common for all sections of a class. The "materials used", "criteria used", "rating" columns are specific to each section/instructor. The educational objectives for several classes were slightly modified to improve consistency. The forms are at http://www.utdallas.edu/~ntafos/ABET/.

4. For "material used" the relevant assignments, exams, projects, etc. should be indicated and used to evaluate how well each class learning objective is met; for example, "Homework 1, Exam 1 - Problem 3". In general, be as specific as is appropriate (keep in mind that you will need the grades for each part you specify in order to compute the "rating"). ABET requires that the item(s) under "materials used" be specific to the educational objective measured (e.g., you can not use the grade for an exam to measure a particular objective unless the whole exam is about that particular objective).

5. For "Criteria Used" identify how your grading scale maps to ABET's 1-5 scale. For example: A=5, B=4, C=3, D=2, F=1 or 85-100=5, 70-85=4, 50-70=3, 35-49=2, 0-34=1, etc. The mapping should be clear, e.g., B = 3,4 is NOT clear. The most common mapping used seems to be A = 5, B = 4, C = 3, D = 2, F = 1 (we suggest you use this as the default and change it only if there is compelling reason to do so).

6. For "ratings" you should compute a class average for each learning objective using the appropriate materials/grades/mapping (i.e., the "material used", "criteria used" columns plus your grades for the homeworks/exams/projects/etc involved. You can weigh each item in "material used" differently as you deem appropriate (an explanatory note should be included).

   - compute the rating to 2 decimal places (avoiding rounding to integers in the preliminary computations as well);
   - include only students that completed the class;
   - compute individual student ratings and then compute the class average (rather than computing class averages and then mapping to the ABET scale).

   To more effectively track students vis-a-vis the ABET requirements, certain classes are designated for more detailed measurements as follows:

CS 2336	SE 3306
CS 3305	SE 4381
CS/SE 3345	SE 4485
CS/SE 3354	ECS 3390
CS/SE 4348	ISSS 3360
CS 4349	

   If you are teaching a section of one of these, you should turn in a detailed spreadsheet with the raw scores for each student.

7. Keep in mind that an accurate assessment is important and that these forms do not exist in their own universe but are only a part of several feedback loops involving comparisons of your assessment with assessments by students, as well as assessments in exit surveys, assessments by employers, advisory committees, focus groups, etc. Continuous evaluation, feedback, and adjustment are at the heart of the ABET accreditation effort.

8. Once completed, the ABET forms should be sent to Simeon Ntafos and Gopal Gupta and your course coordinator. Email the spreadsheet(s) – follow the existing format.

9. In addition to the forms, each instructor must prepare a class file that includes copies of the syllabus, all assignments and exams, samples of student work (illustrating what is typical for each grade – or "excellent", "average", "not satisfactory" work at the very least). Class files will be maintained in ES 4.803. You can turn them in to Simeon or Gopal or the CS staff (but clearly identify them as ABET Materials). We plan to have materials available for the reviewers on line. So, email us anything you have in electronic form as well.

For more information on ABET go to http://www.abet.org

APPENDIX VIII: UNDERGRADUATE CATALOG: COMPUTER SCIENCE

2004 - 2006 Undergraduate Catalog

This page contains revisions since the catalog's original publication

Computer Science (B.S.)

Faculty

Professors: Farokh Bastani, Ramaswamy Chandrasekaran, András Faragó, Gopal Gupta, Dung T. Huynh, Dan Moldovan, Simeon C. Ntafos, Balaji Raghavachari, Hsing-Mean (Edwin) Sha, Ivan H. Sudborough, Bhavani Thuraisingham, Klaus Truemper, Si-Qing Zheng
Associate Professors: Sergey Bereg, Lawrence Chung, Galigekere R. Dattatreya, Sanda Harabagiu, Vasileios Hatzivassiloglou, Jason Jue, Rym Mili, Ivor P. Page, B. Prabhakaran, Ravi Prakash, Haim Schweitzer, S. Venkatesan, Yuke Wang, W. Eric Wong, I-Ling Yen, Kang Zhang
Assistant Professors: Joao Cangussu, Jorge A. Cobb, Kendra M.L. Cooper, Ovidiu Daescu, Jing Dong, Latifur Khan, Neeraj Mittal, Vincent Ng, Kamil Sarac, R.N. Uma, Weili Wu, Youtao Zhang
Senior Lecturers: Tim Farage, Herman Harrison, Sam Karrah, Lawrence King, Rafael Lacambra, Joseph Leubitz, Greg Osbirn, David Russo, Martha Sanchez, Cort Steinhorst, Anthony Sullivan, Laurie Thompson, Nancy Van Ness, Wei Wei

The Computer Science Department offers the B.S. degree in Computer Science and the B.S. degree in Software Engineering. Both are based on a solid foundation of mathematics, including calculus, linear algebra, and discrete mathematics. These programs of study are designed to offer students opportunities to prepare for an industrial, business, or governmental career in a rapidly changing profession and to prepare for graduate study in a field in which further education is strongly recommended. The two programs have the same basis in core computer science, including the analysis of algorithms and data structures, modern programming methodologies, and the study of operating systems. The Computer Science program continues with courses in advanced data structures, programming languages, telecommunications networks, and automata theory, while the Software Engineering program include courses in requirements engineering, software validation and testing, and software architecture, culminating in a challenging project course in which students must demonstrate use of software engineering techniques. Both programs offer a rich choice of elective studies, including courses in artificial intelligence, computer graphics, databases, and compiler design.

The school offers a "fast track" B.S./M.S. option; see Fast Track

Baccalaureate/Master's Degree Program.

Mission of the Department of Computer Science

The mission of the Department of Computer Science is to prepare undergraduate and graduate students for productive careers in industry, academia, and government by providing an outstanding environment for teaching, learning, and research in the theory and applications of computing. The Department places high priority on establishing and maintaining innovative research programs to enhance its education quality and make it an important regional, national, and international resource center for discovering, integrating, and applying new knowledge and technologies.

Goals for the Computer Science Program

The undergraduate Computer Science program is committed to provide students with a high-quality education and prepare them for long and successful careers in industry and government.

Our graduates, while eminently ready for immediate employment, will also be fully ready for focused training as required for specific positions in Computer Science and closely related areas. Graduates interested in highly technical careers, research, and/or academia will be fully prepared to further their education in graduate school.

Educational Objective for the Computer Science Program

On completion of the BS program, students will:

1. have a comprehensive general education background;
2. have solid knowledge in fundamental areas of Computer Science;
3. have the ability to apply their knowledge to the solution of practical and useful problems;
4. have the ability to communicate effectively and work collaboratively;
5. be able to become successful professionals and, if they desire, be able to pursue graduate study;
6. recognize the need for lifelong learning and have the knowledge and skills that prepare them to adapt to rapid technological changes; and
7. understand the ethical, societal, and global issues associated with the computing field.

Bachelor of Science in Computer Science Degree Requirements (121 hours)

I. Core Curriculum Requirements1: 42 hours

A. Communication (6 hours)
- 3 hours Communication (RHET 1302)
- 3 hours Professional and Technical Communication (ECS 3390) [2]

B. Social and Behavioral Sciences (15 hours)
- 6 hours Government (GOVT 2301 and 2302)
- 6 hours American History
- 3 hours Social and Behavioral Science (ISSS 3360)

C. Humanities and Fine Arts (6 hours)
- 3 hours Fine Arts (ARTS 1301)
- 3 hours Humanities (HUMA 1301)

D. Mathematics and Quantitative Reasoning (6 hours)
- 6 hours Calculus (MATH 2417 and 2419)[3]

E. Science (9 hours)
- 6 hours Lecture courses (PHYS 2325 and 2326) [4]
- 2 hours Laboratory courses (PHYS 2125 and 2126) [4]
- 4 hours Science Elective[4]

[1] Curriculum Requirements can be fulfilled by other approved courses from accredited institutions of higher education. The courses listed in parentheses are recommended as the most efficient way to satisfy both Core Curriculum and Major Requirements at U.T. Dallas.

II. Major Requirements: 58 hours

Major Preparatory Courses (22 hours beyond Core Curriculum)
- CS 1337 Computer Science I
- CS 2110 Introduction to Digital Systems Laboratory
- CS 2305 Discrete Mathematics for Computing I
- CS 2310 Introduction to Digital Systems
- CS 2336 Computer Science II
- MATH 2417 Calculus I[3]
- MATH 2418 Linear Algebra
- MATH 2419 Calculus II[3]
- PHYS 2125 Physics Laboratory I[4]
- PHYS 2126 Physics Laboratory II[4]
- PHYS 2325 Mechanics and Heat[4]
- PHYS 2326 Electromagnetism and Waves[4]
- 4 hours Science Elective[4]

Major Core Courses (27 hours beyond Core Curriculum)
- CS 3305 Discrete Mathematics for Computing II
- CS/SE 3341 Probability and Statistics in Computer Science
- CS/SE 3345 Algorithm Analysis and Data Structures
- CS/SE 3354 Software Engineering
- CS 4337 Organization of Programming Languages
- CS/SE 4340 Computer Architecture
- CS/SE 4348 Operating Systems Concepts

CS 4349 Advanced Data Structures and Algorithms
CS 4384 Automata Theory
ECS 3390 Professional and Technical Communication[2]
ISSS 3360 Politics and Values in Business and Technology[5]

Major Guided Electives (9 hours)

CS guided electives are 4000 level CS courses approved by the student's CS advisor. The following courses may be used as guided electives without the explicit approva of an advisor:

CGS/CS 4314 Intelligent Systems Analysis
CGS/CS 4315 Intelligent Systems Design
CGS/CS 4352 Human Computer Interaction I
CGS/CS 4353 Human Computer Interaction II
CS 4334 Numerical Analysis
CS 4336 Advanced Java Programming
CS/SE 4347 Database Systems
CS 4361 Computer Graphics
CS 4365 Artificial Intelligence
CS/SE 4376 Object-Oriented Programming Systems
CS 4380 Senior Design Project
CS 4386 Compiler Design
CS/TE 4390 Computer Networks
CS 4391 Introduction to Computer Vision
CS 4392 Computer Animation
CS 4393 Computer and Network Security
CS 4394 Implementation of Modern Operating Systems
CS 4396 Networking Laboratory
CS 4397 Embedded Computer Systems
CS/SE 4399 Senior Honors in Computer Science/Software Engineering
EE 4325 Introduction to VLSI Design
EE 4420 Microprocessor Systems Design
SE 4351 Requirements Engineering
SE 4352 Software Architecture and Design
SE 4367 Software Testing, Verification, Validation and Quality Assurance
SE 4381 Software Project Planning and Management
SE 4485 Software Engineering Project

[2] Hours fulfill the communication elective of the Core Curriculum.
[3] Six hours of Calculus are counted under Mathematics Core, and two hours of Calculus are counted as Major Preparatory Courses.
[4] Nine hours of Science are counted under Science Core. Three hours are counted under Major Preparatory Courses. Students should consult an advisor for specific classes that satisfy this requirement.
[5] Hours contribute to the Social and Behavioral Sciences component of the Core

Curriculum.

III. Elective Requirements: 21 hours

Advanced Electives (6 hours)

All students are required to take at least six hours of advanced electives outside their major field of study. These must be either upper-division classes or lower-division classes that have prerequisites.

Free Electives (15 hours)

All students must accumulate at least 120 hours of university credit to graduate. Both lower-and upper-division courses may count as free electives but students must complete at least 51 hours of upper-division credit to qualify for graduation. Not all courses offered by the University can be used as a free elective. Please consult with your advisor.

At least 9 hours of electives (out of the 21 hours required) must be outside Science and Engineering. At least 6 hours (out of the designated 21) should be in Humanities, Arts, and other areas that broaden the student's educational experience. Consult an advisor for specific classes.

Fast Track Baccalaureate/Master's Degrees

In response to the need for post-baccalaureate education in the exciting field of computer science, a Fast Track program is available to exceptionally well-qualified students who choose their courses carefully. At the end of five years of successful study, it is possible to earn both the B.S. and the M.S. degrees in Computer Science (or M.S. in Computer Science with Major in Software Engineering). Being within 30 hours of graduation, a student admitted to the graduate program and accepted into the Fast Track program may, during the senior year, take 15 graduate hours that may be used to complete the baccalaureate degree and also to satisfy requirements for the master's degree.

Interested students should see the Associate Dean of Undergraduate Education (ADU) for specific admission requirements to the Fast Track program.

Honors Program

The Department of Computer Science offers upper-division Honors for outstanding students in both the B.S. in Computer Science and B.S. in Software Engineering degree programs. These programs offer special sections of designated classes and other activities designed to enhance the educational experience of exceptional students. Admission to the Honors programs requires a 3.5 or better GPA in at least

30 hours of coursework. Graduation with Honors requires a 3.5 or better GPA and completion of at least 6 honors classes, including a Senior Thesis or Senior Design Project class. For more details, contact the Office of Undergraduate Advising (ECS South 2.502; (972)883-2004).

Minors

A minor in Computer Science requires 21 credit hours earned through the following courses:

CS 1337 Computer Science I
CS 2305 Discrete Mathematics for Computing I
CS 2336 Computer Science II
CS 3305 Discrete Mathematics for Computing II
CS/SE 3345 Data Structures and Introduction to Algorithms
CS/SE 3354 Software Engineering
CS 43XX Elective (any 4000-level organized class)

APPENDIX IX: COMPUTER SCIENCE DEGRE PLAN FORM

B.S. in COMPUTER SCIENCE (2004-2006 Catalog)

Student ______________________ Advisor ______________________

Social Security Number _____-_____-_____ Date of Review ______________________

Student Email______________________ Student Phone ______________________

MAJOR PREPARATORY REQUIREMENTS

LD	UD	NOTES	COURSE TITLE	COURSE #	GRADE	SEM	INFO
3			Computer Science I	CS 1337			
3			Discrete Mathematics I	CS 2305			
3			Computer Science II	CS 2336			
4			Linear Algebra	Math 2418			
3			Intro to Digital Systems	CS 2310			
1			Intro to Digital Systems Lab	CS 2110			
4		2	Calculus I	Math 2417			
4		2	Calculus II	Math 2419			
3		2	Mechanics and Heat	Phys 2325			
1		2	Mechanics and Heat Lab	Phys 2125			
3		2	Electricity and Magnetism	Phys 2326			
1		2	Electricity and Magnetism Lab	Phys 2126			
4		1,2	Science Elective				
37							

MAJOR CORE REQUIREMENTS*

*(*Courses in this block used to calculate major GPA)*

LD	UD	NOTES	COURSE TITLE	COURSE #	GRADE	SEM	INFO
	3		Discrete Math for Computing II	CS 3305*			
	3		Probability & Statistics in Comp Sci	CS/SE 3341*			
	3		Algorithm Analysis & Data Struc	CS/SE 3345*			
	3		Software Engineering	CS/SE 3354*			
	3		Organization of Programming Lang	CS 4337*			
	3		Computer Architecture	CS/SE 4340*			
	3		Operating Systems Concepts	CS/SE 4348*			
	3		Adv Data Structures & Algorithms	CS 4349*			
	3		Automata Theory	CS 4384*			
	3		CS Guided Elective	CS 43XX*			
	3		CS Guided Elective	CS 43XX*			
	3		CS Guided Elective	CS 43XX*			
	3		Professional & Technical Comm	ECS 3390*			
	3		Politics & Values in Bus & Tech	ISSS 3360*			
	42						

REMAINING CORE CURRICULUM

LD	UD	NOTES	COURSE TITLE	COURSE #	GRADE	SEM	INFO
3			Rhetoric	Rhet 1302			
3			Const & Pol Behavior, US & TX	Govt 2301			
3			Political Institutions, US & TX	Govt 2302			
3		3	Themes & Ideas in Am Hist	Hst 1301			
3		3	Issues in American History	Hst 2301			
3		3	Exploration of the Humanities	HUMA 1301			
3		3	Exploration of the Arts	ARTS 1301			
21							

ELECTIVES

LD	UD	NOTES	COURSE TITLE	COURSE #	GRADE	SEM	INFO
	3	4,7	Advanced Free Elective				
	3	4,7	Advanced Free Elective				
	3	5,7	Free Elective				
3		5,7	Free Elective				
3		5,7	Free Elective				
3		5,7	Free Elective				
3		5,7	Free Elective				
1		6	*Oral Communication & Critical Thinking*	*Rhet 1101*			
12	9						
70	51	***Total Hours***	***121 hrs required for graduation, min 51 must be Upper Division***				

NOTES

1. Any LD or UD Physical Science Course; see advisor for details.	5. May be taken LD (Lower Division) or UD (Upper Division)
2. Course meets both major and general education requirement.	6. Waived if more than 1 yr since high school graduation.
3. Other course options available; see advisor for details.	7. At least 6 hrs of electives must in humanities or other broadening field.
4. Must be outside major and be Upper Division or require pre-req	7. Consult your advisor for specific courses.

Revision date – 5/12/05

APPENDIX X

FACULTY VITAE

NAME:	Farokh B. Bastani
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of California, Berkeley, 1980 MS, Computer Science, University of California, Berkeley, 1978 B.Tech., Electrical Engineering, Indian Institute of Technology, Bombay, 1977
YEARS IN SERVICE AT UT DALLAS:	8 years
RELATED EXPERIENCE:	Director, Embedded Software Center, UTD, 2000-present Professor, Department of Computer Science, University of Houston, 1993-1997 Visiting Scholar, Department of Computer Science, Michigan State University, 1993-1994 Associate Professor, Department of Computer Science, University of Houston, 1986-1993 Visiting Scholar, Department of Electrical Engineering and Computer Science, University of California, Berkeley, 1986-1987 Assistant Professor, Department of Computer Science, University of Houston, 1980-1986 Research Assistant, Electrical Research Laboratory, University of California, Berkeley, 1978-1980
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	D. Wang, F.B. Bastani, & I.L. Yen, "A systematic design method for high quality process-control systems development," *Int. J. of Softw. Eng. and Know. Eng.* (*JSEKE*), Vol. 14, No. 1, 2004, pp. 43-59. F. Luo, L. Khan , F. Bastani, I-Ling Yen and J. Zhou, "A dynamically growing self-organizing tree (DGSOT) for hierarchical clustering gene expression profiles," *Bioinformatics Journal*, Oxford University Press, UK., Vol. 20, No. 16, 2004, pp. 2605-2617. S. Kim, F.B. Bastani, I.-L. Yen, and I.-R. Chen, "Systematic reliability analysis of a class of application-specific embedded software frameworks," *IEEE Trans. on Software Eng.*, Vo. 30, No. 4, Apr. 2004, pp. 218-230. M. Gupta, M. Tu, L. Khan, F. Bastani, and I.-L. Yen, "A study of the model and algorithms for handling location dependent continuous queries," To appear in *Knowledge and Information Systems: An Intl. Journal*, Springer-Verlag London.
PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE Computer Society ACM
HONORS & AWARDS:	IEEE Computer Society Meritorious Service Certificate, 2001 IEEE Computer Society Golden Core Award, 1996

	IEEE Computer Society Certificate of Appreciation, 1995 IEEE Computer Society Meritorious Service Certificate, 1992
COURSES TAUGHT 2001-2004:	CS 6354: Advanced Software Engineering (Fall'01, Spring'02, Fall'02, Spring'03) CS 7301: AI-Based Software Engineering (Fall'03) CS 6396: Real-Time Systems (Spring'04, Fall'04)
OTHER ASSIGNED DUTIES:	Coordinator, Software Engineering Group, 2002-2004 Faculty Search Committee 1997-2001 Third Year Review Committee, 2003-2004 Tenure Review Committee, 1998-2001 Equipment Committee 2001-2003 Department Bylaws Committee, 2003-2004 ECS Research Committee, 1999-2001 ECS Post Tenure Review Committee, 2003-2004
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Editor-in-Chief, IEEE Transactions on Knowledge and Data Engineering, 1997-2000 Editor for the International Journal on Artificial Intelligence Tools, 1993-present Editor for the *International Journal of Knowledge and Information Systems*, 1999-present Co-Guest Editor (with Y. Deng) of *International Journal of Software* *Engineering and Knowledge Engineering* special issue on Embedded Computer Systems, April 2002 Steering Committee Member, *IEEE Symposium on High Assurance Systems Engineering*, 1996-present

NAME:	Sergey Bereg
DATE OF BIRTH:	December 29, 1962
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, Minsk Institute of Mathematics. 1992 MS Computer Science, Ural State University, 1985 BS, Computer Science and Mathematics, Ural State University, 1983
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Visiting Professor, Duke University, 2001-2002 Research Associate, University of British Columbia, 1998-2001 Senior Programmer, Uraltransbank, 1994-1998 Senior Scientist, Ural State University, 1994-1998 Assistant Professor, Ural State University, 1990-1994 Teaching Assistant, Ural State University, 1989-1990 Senior Engineer, Research Institute of Automation, 1985-1989
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	S. Bereg, B. Bhattacharya, D. Kirkpatrick, & M. Segal, “Competitive Algorithms for Mobile Centers”, *Special Issue of the Journal on Special Topics in Mobile Networking and Applications (MONET) on "Foundations of Mobile Computing", 2005.* A. Ban, S. Bereg, & N. Mustafa, “On a conjecture of Wiener Indices in Combinatorial Chemistry”, *Algorithmica*, 2004 S. Bereg, “Cylindrical Hierarchy for Deforming Necklaces”, *International Journal of Computational Geometry and Applications*, 2004 M.J. Spriggs, J.M. Keil, S. Bespamyatnikh, M. Segal, & J. Snoeyink, “Approximating the geometric minimum-diameter spanning tree”, *Algorithmica*, 2004 S. Bespamyatnikh, “Computing homotopic shortest paths in the plane,” *Journal of Algorithms*, 2003
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Association of Computing Machinery (ACM)
HONORS & AWARDS:	Gold Medal at the National Mathematical Competition, 1979 Silver Medal at the National Mathematical Competition, 1980
COURSES TAUGHT 2001-2004:	CS 3345 Algorithm Analysis and Data Structures CS 5343 Algorithm Analysis and Data Structures

	CS 6363 Design and Analysis of Computer Algorithms CS 6V81 Computational Biology and Geometry
OTHER ASSIGNED DUTIES:	Member, Search Committee. Member, Admission Committee.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	João W. Cangussu
DATE OF BIRTH:	June 24, 1969
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Sciences, Purdue University, 2002 MS, Computer Sciences, University of São Paulo, 1993 BSD, Computer Science, Federal University of Mato Grosso do Sul, 1990
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Teaching Assistant, Computer Sciences, Purdue University, 1998-2001 Assistant Professor, Computer Science, Federal University of Mato Grosso do Sul, 1993-1997
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	J.W. Gangussu, R.A. DeCarlo, * A.P. Mathur, "Using sensitivity analysis to validate state variable model of the software test process," *IEEE Transactions on Software Engineering*, 2003 J.W. Gangussu, R.A. DeCarlo, * A.P. Mathur, "A formal model for the software test process," *IEEE Transactions on Software Engineering*, 2002 J.W. Cangussu, "A software test process stochastic control model based on CMM characterization," *Software Process: Improvement and Practice*, 2004. J.W. Cangussu, P.C. Masiero, & J.C. Maldanado, "Programmed execution of statecharts," *Brazilian Computer Journal*, 1994 J.W. Cangussu, R. Penteado, P.C. Masiero, & J.C. Maldonado, "Validation of statecharts based on programmed execution," *Journal of Computing and Information*, 1995
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Association for Computing Machinery(ACM) IEEE Computer Society
HONORS & AWARDS:	Maurice Halstead memorial Award for Outstanding Research in Software Engineering, Purdue University, 2001. TA Award, Purdue University, 2001.
COURSES TAUGHT 2001-2004:	Software Process Modeling, Simulation, and Control Software Testing, Validation, Verification and Quality Assurance
OTHER ASSIGNED DUTIES:	CS Undergraduate Curriculum Committee, 2004.

SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:

NAME:	R. Chandrasekaran
DATE OF BIRTH:	
ACADEMIC RANK:	Ashbel Smith Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph. D. in Operations Research, University of California, Berkeley, 1967. B. Tech. (Hons with Distinction) in Mechanical Engineering, Indian Institute of Technology, Bombay, India, 1963.
YEARS IN SERVICE AT UT DALLAS:	30 years
RELATED EXPERIENCE:	Interim Dean, Computer Science, UT Dallas, 2002-2003. Associate Professor, Operations Research, Case Western Reserve University, 1972-75. Assistant Professor, Operations Research, Case Western Reserve University, 1969-72. Instructor, University of California, Berkley, 1967. Visiting Appointments: Visiting Professor, Tel Aviv University, 1981. Visiting Professor, School Of Management, Northwestern university, 1978-79.
CONSULTING, PATENTS, ETC.:	Consultant, Yottanetworks, 2000-02. Consultant, Qtera/Nortel, 1997-2000. Consultant, BNR/Nortel, 1995-99. Consultant, MCU, 1995-97. Consultant Banking Systems International Inc., Reno, Nevada, 1972-75. Scientist/Consulting Staff, Optimum Systems Inc., Palo Alto, 1967-69.
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	"Computational Complexity of Integrated Models of Network Design and Facility Location" (with J. Bhadury and L. Gewali), *Southwest Jour. of Pure and Applied Mathematics*, July 2000, pp. 30-43. "Maximizing Residual Flow under Arc Destruction" (with Y.P. Aneja and K.P.K. Nair), *NETWORKS*, 38, #4, (2001), pp. 194-198. "Using Linear Programming in a Business-to-Business Auction Mechanism" (with Milind Dawande and J. Kalagnanam), *Review of Marketing Science*, 1, #4, July 2002. "Parametric Min-Cuts Analysis in a Network", Y.P. Aneja, R. Chandrasekaran, K.P.K. Nair, *Discrete Applied Mathematics*, 127 (2003), 679-689. "Parametric Overall Min-cut Trees", Y.P. Aneja, R.

Chandrasekaran, K.P.K. Nair), *Information Processing Letters*, 85 (2003), 105-109.
"Improved Bounds for the On-line Scheduling Problem", John Rudin, R. Chandrasekaran, *SIAM Journal on Computing*, 32, (2003), 717-735.
"Scheduling Multiple Parts in a Robotic Cell Served by a Dual Gripper Robot", C. Sriskandarajah, I. Drobouchevitch, S.P. Sethi, R. Chandrasekaran, *Operations Research*, 52 (2004), 65-82.
"Minimum Cost System Reliability with Discrete Choice Sets for Components", Y.P. Aneja, R. Chandrasekaran, K.P.K. Nair, *IEEE Transactions on Reliability* 53 (2004), 71-76.
"Prognosis Using an Isotonic Prediction Technique", Y. Ryu, R. Chandrasekaran, V. Jacob, *Management Science*, 50, #6, (June 2004), pp. 777-785.
"Multi-path Multi-terminal Flow Synthesis", R. Chandrasekaran, K.P.K. Nair, Y.P. Aneja, S.N. Kabadi, *Discrete Applied Mathematics* 143 (2004), pp. 182-193.
"Isotonic Separation", R. Chandrasekaran, Young Ryu, Varghese Jacob, Sung-Chul Hong, *INFORMS Journal on Computing* (accepted)

SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Mathematical Programming Society Operations Research Society of India.
HONORS & AWARDS:	Ashbel Smith Professor, UT Dallas, 1997-present. Nominated for the George B. Dantzig Prize in Mathematical Programming, 1988. Polykarp Kusch Lecturer, 1986-87. Gurdas Chatterjee Award, Operations Research Society of India, 1983. Nominated for Piper Teaching Award from UT Dallas, 1980. Merit Scholarship (given to top 10 students at IIT, Bombay) 1960-62.
COURSES TAUGHT 2001-2004:	CS 4349: Advanced Data Structures and Algorithms CS 6363: Design and Analysis of Computer Algorithms CS 6381: Combinatorics and Graph Algorithms CS 6V81: Scheduling
OTHER ASSIGNED DUTIES:	Faculty Senate Academic Council Presidential Search Committee Committee on Qualifications (Vice-Chair) CS Advisory Committee PhD Committee(Chair) Faculty Search Committee. TA Assignment Committee.

SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:

NAME:	Lawrence Chung
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of Toronto, 1993 M.Sc., Computer Science, University of Toronto, 1984 B.Sc., Hons, Computer Science (Data Management), University of Toronto, 1981
YEARS IN SERVICE AT UT DALLAS:	12 years
RELATED EXPERIENCE:	Visiting Scholar, Center for Strategic Technology Research, Andersen Consulting, 1994 Lecturer, Computer Science, University of Toronto, 1993-1994 Software Developer, Dept. of Computer Science, University of Toronto, 1984-1986
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	L. Chung and K. Cooper, "Matching, Ranking, and Selecting Components: A COTS-Aware Requirements Engineering and Software Architecture Approach," *Proc. Intl Workshop on Models and Processes for the Evaluation of COTS Components (MPEC'04)*, May 25, 2004, Edinburgh, Scotland. L. Chung & N. Subramanian, "Adaptable Architecture Generation for Embedded Systems," *Special Issue on Computer Systems, Journal of Systems and Software* (forthcoming). L. Chung & K. Cooper, "Defining Goals in a COTS aware requirements engineering approach," *Systems Engineering:* **7**(1), 2004. pp. 61-83. L. Chung & Narayanan, "Architecture-based semantic evolution of embedded systems: a study of remotely controlled systems," *Journal of Software Maintenance and Evolution*: 15(2), May/June, 2003. L. Chung, K. Cooper, A. Yi, "Developing adaptable software architectures using design patterns: an NFR approach," *Computer Standards and Interfaces (CS&I)*, 25(3), 2003. pp. 253-260. L. Chung, B. A. Nixon, E. Yu and J. Mylopoulos, "Non-Functional Requirements in Software Engineering", *Kluwer Academic Publishing*, 2000. 472pp.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A	Institute of Electrical and Electronic Engineers.(IEEE)

MEMBER:	
HONORS & AWARDS:	Achievement Award, World Academy of Sciences, 2002, 2003, 2004
COURSES TAUGHT 2001-2004:	Object Oriented Analysis and Design Software Architecture and Design Requirements Engineering Software Engineering. Senior Design Project
OTHER ASSIGNED DUTIES:	Graduate Curriculum Committee, 2003. Admissions Committee, 1997-04. Ph.D. Qualifying Examination Committee.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Editorial Board Member, Requirements Engineering, International Journal, 1998-present. Program Co-Chair, Sixth International Conference on Software Engineering. Artificial Intelligence, Networking and Parallel/Distributed Computing. NSF Panel Reviewer, 2004. Program Co-Chair, Int. Workshop on Requirements Engineering on COTS.

NAME:	Jorge Arturo Cobb
DATE OF BIRTH:	April 15, 1966
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Sciences, University of Texas at Austin, 1996 MA, Computer Sciences, University of Texas at Austin, 1989 BS, Computer Science, University of Texas at El Paso, 1987
YEARS IN SERVICE AT UT DALLAS:	7 years
RELATED EXPERIENCE:	Assistant Professor, Computer Science, University of Houston, 1995-1998 Technical staff, AT&T Bell Laboratories, 1989-1990, 1993-1994 Teaching Assistant, Computer Science, University of Texas at Austin, 1991-1992
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	J. Cobb, M. Gouda, & D. Sidhu, "Hello Again: Convergence of the Hello Protocol in OSPF", accepted for publication, *Journal of High-Speed Networks*, IOS Press. J. Cobb, "Scalable Quality of Service across Multiple Domains", accepted for publication, *Computer Communications*, Elsevier. J.Cobb & M. Lin, "The timely-token protocol," *Computer Communications*, 2004 J. Cobb & M. Gouda, "Stabilization of general loop-free routing," *Journal of Parallel and Distributed Computing*, 2002 J. Cobb, "Preserving quality of service guarantees in spite of flow aggregation," *IEEE/ACM Transactions on Networking*, 2002
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE
HONORS & AWARDS:	
COURSES TAUGHT 2001-2004:	CS 4390 Computer Networks CS 5390 Computer Networks CS 6390 Advanced Computer Networks
OTHER ASSIGNED DUTIES:	Member of Networking Group, Graduate Curriculum Committee, actively involved in academic conference organization
SPECIFIC PROGRAMS IN WHICH INVOLVED TO	

IMPROVE TEACHING &
PROFESSIONAL
COMPETENCE:

NAME:	Kendra M.L. Cooper
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Electrical & Computer Engineering, University of British Columbia, 2001 MASc Electrical & Computer Engineering, University of British Columbia, 1995 BASc, Electrical & Computer Engineering, University of British Columbia, 1993
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Senior Systems Engineer, Motorola Canada, Network Solutions Sector, 2000 Requirements Engineer, Hughes Aircraft of Canada, 1991, 1993, 1997. Sessional Instructor, Computer Science, University of British Columbia, 1998-2000
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	L. Chung, K. Cooper, A. Yi, "Developing Adaptable Architectures for Real-time Systems Using Design patterns," *Computer Standards & Interfaces, 2003* L. Chung, K. Cooper, A. Yi, "Towards Adaptable COTS-Aware Software Architecting," *Computer Standards & Interfaces, 2003* L. Chung & K. Cooper, "Defining Goals in a COTS-Aware Requirements Engineering Approach," *Software Engineering Journal, 2004* K. Cooper, L. Dai, & Y. Deng, "Modeling and Analysis of Performance Aspects: a UML Based Design Approach," *System and Software Architectures of the Journal of Science of Computer Programming* (accepted) E. Wong, Y. Qi, & K. Cooper, "Source Code-based Software Risk Assessment," *ACM Symposium on Applied Computing (SAC 2005)*
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	International Council on Systems Engineering
HONORS & AWARDS:	Nominated for Teacher of the Year Award, 2001-2002 National Science & Engineering Research Council Graduate Scholarship, 1995

	Simons Foundation Scholarship, 1995 British Columbia Advanced Science Institute Graduate Recruitment Scholarship, 1995 Center of Integrated Computer Science Research Top-up Award, 1995 Killam Pre-doctoral Fellowship, 1998-1999
COURSES TAUGHT 2001-2004:	CS 2305 Discrete Mathematics I SE 3306 Mathematical Foundations for Software Engineers CS 4485 Software Engineering Project SE 6354 Advanced Software Engineering CS 6389 Formal methods and Programming Methodologies
OTHER ASSIGNED DUTIES:	University Library Committee, 2002-2004 Computer Science Dept. Library Committee, 2002-2004 Computer Science Dept. Equipment Committee, 2004 Qualifying Examination Committee for CS 6354, 2002-2004 Qualifying Examination Committee for CS 6361, 2002-2004 ABET Accreditation Working Group IBM Rational Corp. Scholar Program Representative, 2002-2004
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Member, Program Committee for Model Based Requirements Engineering Workshop, 2001 Member, Program Committee for IEEE Conference on Computer Software and Applications (COMPSAC), 2003-2005 Member, IEEE-CS/ACM Computing Curricula Software Engineering Joint Task Force on Computing Curricula. Chair for the Committee on Introductory Modules and Courses. Member, Program Committee, Workshop on Adaptable Software Architectures, 2002-2004 Member, Program Committee, Workshop on Models and Processes for the Evaluation of COTS Components (MPEC 2004) Member, Program Committee, Workshop on International Workshop on Requirements Engineering for COTS Components (RECOTS 2004) Member, Program Committee, IEEE International Symposium on Multimedia Software Engineering, (MSE 2004) Local Chair, IEEE Symposium on Visual Languages and Human-Centric Computing (HCC 2005)

NAME:	Ovidiu Daescu
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science and Engineering, University of Notre Dame, 2000 MS, Computer Science and Engineering, University of Notre Dame, 1997 MS/MS, Engineering/Computer Science and Automation, Technical Military Academy (Romania), 1991
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Postdoctoral Research Associate, University of Notre Dame, 2000 Research and Teaching Assistant, University of Notre Dame, 1995-2000 Teaching Lecturer, Technical Military Academy, 1992-1995 Programmer/Analyst, Technical Military Academy, 1991-1992
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	O. Daescu and N. Mi, “Polygonal Path Approximation: A query based approach”, *Computational Geometry: Theory and Applications* 30 1 (2005) 41-58. O. Daescu and J. Luo, “Cutting out Polygons with Lines and Rays”, *Proc. 15th International Symposium on Algorithms and Computation* (2004) 669-681. O. Daescu, “New Results on Path Approximation”, *Algorithmica* 38 2 (2003) 131-143. D. Z. Chen, O. Daescu, X. Hu and J. Xu, “Finding an Optimal Path without Growing the Tree”, *Journal of Algorithms*, 49 1 (2003) 13-41. D. Z. Chen and O. Daescu, “Space Efficient Algorithms for Approximating Polygonal curves in two Dimensional Space”, *Int. Journal of Computational Geometry and Applications* 13 2 (2003) 95-111. D.Z. Chen, O. Daescu, X. Hu, X. Wu, & J. Xu, “Determining an optimal penetration among weighted regions in two and three dimensions,” *Journal of Combinatorial Optimization*, 2001 D.Z. Chen, O. Daescu, & K.S. Klenk, “On geometric path query problems,” *International Journal of Computational Geometry and Applications*, 2001 S. Emilda, L. Jacob, O. Daescu, & B. Prabhakaran, “Flexible disk scheduling strategies for multimedia presentation servers,”

	Multimedia Tools and Applications, 2003 D.Z. Chen, O. Daescu, Y. Dai, N. Katoh, X, Wu, & J. Xu, “Efficient algorithms and implementations for opt6imizing the sum of linear fractions with applications,” *Journal of Combinatorial Optimization*, 2003.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Association for Computing Machinery and ACM-SIGACT.
HONORS & AWARDS:	Graduate Student Fellow of the Center for Applied Mathematics, University of Notre Dame. 1998-99. Fellowship from the Center for Applied Mathematics, University of Notre Dame, 1998-99. Prizes in ‘Traian Lalescu’ Romanian national Competition in Physics (1987) and Mathematics (1987, 1988) and in National Programming Contest (1989).
COURSES TAUGHT 2001-2004:	CS 3345 Algorithm Analysis and Data Structures CS 4349 Advanced Algorithms and Data Structures CS 6363 Design and Analysis of Computer Algorithms CS 6V81 Geometric Optimization CS 7301 Recent Advances in Computing: Applied Algorithms
OTHER ASSIGNED DUTIES:	Computer Science, Master Research Track Committee. UTD Committee on Biotechnology.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Co-Organizer, Third International Workshop on Computational geometry and Applications, Montreal, Canada, May 2003. NSF Panel Review Member, May, June 2004.

NAME:	G. R. Dattatreya
DATE OF BIRTH:	October 18, 1954
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, School of Automation, Indian Institute of Science (Bangalore), 1981 ME, Electrical Communication Engineering, Indian Institute of Science (Bangalore), 1977 B.Tech, Electronics and Communication Engineering, Indian Institute of Technology (Madras), 1975.
YEARS IN SERVICE AT UT DALLAS:	18 Years
RELATED EXPERIENCE:	Visiting Professor, Center for Artificial Intelligence, Monterrey Institute of Technology, Monterrey, Mexico, 1999-2000 Visiting Assistant Professor , Department of Computer Science, University of Maryland, 1983-1985 Scientist, Defense Research and Development Organizations, Delhi, 1981-82
CONSULTING, PATENTS, ETC.:	Training Programmers at Perot Systems Inc., Spring, 1996.
STATE(S) IN WHICH REGISTERED:	Texas
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	S. S. Kulkarni, G. R. Dattatreya, H. Martinez and R. Soto, “Adaptive Control of Heterogeneous Ad-hoc networks”, *Wireless Communications and Mobile Computing* 4 (December 2004) 963-975. G. R. Dattatreya and S. S. Kulkarni, “Performance of Communication networks fielding bursty data traffic”, *Annual Review of Communication* 57 (November 2004) 1259-1273. S. S. Kulkarni and G. R. Dattatreya, “SMART: Statistically multiplexed adaptive routing techniques for adhoc networks”, *Wireless Networks* 10 (March 2004) 89 -101. G. R. Dattatreya and F. Fang, “Parameter Estimation: Known Vector Signals in Unknown Gaussian Noise”, *Pattern Recognition* 36 (2003) 2317-2332. G. R. Dattatreya, “Gaussian Mixture parameter Estimation with Known Means and Unknown Class-dependent variances”, *Pattern Recognition* 35 (July 2002) 1611-1616.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Member, IEEE.

HONORS & AWARDS:	Externally Funded Contracts at UTD: "Network Modeling and Optimization Problems," (with Dr. S. Venkatesan) Alcatel Network Systems; $90,000 for June-Dec. 1992. "Development of a Simulation Model for High Density Communication Network," Electrospace Systems, Inc., Richardson, TX; $30,000, June - Dec. 1994. "A Study of Strategies for IP Quality of Service," (with B. Chen, R. Prakash, 1. L. Yen, and S. Q. Zheng) Alcatel Network Systems, Inc., Richardson, TX; $50,000, Jan. - Dec. 1999.Industrial Research Agreements for Graduate Student Support. Total funds: Approximately $65,000.
COURSES TAUGHT 2000-2005:	CS 5390 Computer Networks CS/SE 4340 Computer Architecture CS 6352 Performance of Computer Systems and Networks
OTHER ASSIGNED DUTIES:	Member, UTD Senate, Fall 2003-present. Member, CS Ph.D. Committee, Fall 2004 – present. CS Graduate Curriculum Committee Chair, Fall 2000 – Summer 2004. CS Faculty Search Committee, Graduate Admissions Committee and Financial Aid Committee.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Jing Dong
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of Waterloo, Canada, 2002 MMath, Computer Science, University of Waterloo, Canada, 1997 BSc, Computer Science, Peking University, 1992
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Teaching & Research Assistant, Dept. of Computer Science, University of Waterloo, Canada, 1995-2002 Software Engineer, Computer Systems Group, University of Waterloo, Canada, 1997 Software Engineer, Database & Multimedia Group, Peking University, 1993-1995 Software Developer, Beida Founder Group (China), 1992-1993
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Jing Dong, Paulo Alencar, and Donald Cowan, "Automating the Analysis of Design Component Contracts", T*he International Journal of Software - Practice and Experience (SPE), Wiley,* 2005, 45 pages (to appear). Jing Dong, Paulo Alencar, and Donald Cowan, "A Behavioral Analysis and Verification Approach to Pattern-Based Design Composition", T*he International Journal of Software and Systems Modeling, Springer-Verlag, Volume 3, Number 4, December 2004, Pages 262-272* Jing Dong, "Adding Pattern related information in structural and behavioral diagrams," *International Journal of Information and Software Technology*, *Elsevier-Science*, Vol. 46, Issue 5, April 2004. 293-300. D.M. Berry, K. Daudjee, Jing Dong, I. Finestein, M. Nelson, T. Nelson, & L. Ou, "Users' Manual as a Requirements Specification: Case Studies," *International Journal of Requirements Engineering*, February 2004, 67-82. Jing Dong, P. Alencar, & D. Cowan, "A behavioral analysis and verification approach to pattern based design composition," *International Journal of Software and Systems Modeling*, 2003 Jing Dong, "UML extensions for design pattern compositions," *International Journal of Object Technology*, 2002

SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Association for Computing Machinery IEEE Computer Society Consortium for Software Engineering Research Center for Information Technology Ontario Canadian Mathematical Society International Society for Professionals in e-Commerce
HONORS & AWARDS:	IBM CAS Fellowship, 2000-2002 ACM Student Travel Scholarship, 2000 CITO Scholarship, 1997-2000 ITRC Scholarship, 1996-1997 International Student Scholarship, 1995-1996
COURSES TAUGHT 2001-2004:	CS 6362 Software Architecture and Design
OTHER ASSIGNED DUTIES:	PhD Exam Qualifying Committee (CS 6362, CS 6388) CS Department Equipment Committee. Chair of CS Department Library Committee.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Certificate in University Teaching, University of Waterloo, Canada, 2001-2002

NAME:	Timothy P. Farage
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	M.S., Computer Science, University of Texas at Dallas B.A., Mathematics Education and Psychology, Case Western Reserve University
YEARS IN SERVICE AT UT DALLAS:	14 years
RELATED EXPERIENCE:	Senior Software Engineer, INET TECHNOLOGIES, Inc. Software Engineer, Electronic Systems, RAYTHEON E-SYSTEMS
CONSULTING, PATENTS, ETC.:	Technical Managers Training Certification Communications Skills Course GeoProbe Manager Certification Security Clearance: SSBI
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	Received the ERIK JONSSON SCHOOL OF ENGINEERING AND COMPUTER SCIENCE 'Excellence in Teaching' Award for the 2003-2004 school year
COURSES TAUGHT 2001-2004:	CS 2305 Discrete Mathematics I CS 3305 Discrete Mathematics II CS 4390 Computer Networks CS 5333 Discrete Mathematics
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Andras Farago
DATE OF BIRTH:	June 3, 1952
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Dr. Habil, Technical University of Budapest, 1997 Doctor of the Hungarian Academy of Sciences, 1996 PhD, Electrical Engineering, Technical University of Budapest, 1981 MSc, Electrical Engineering, Technical University of Budapest, 1979 BSc, Electrical Engineering, Technical University of Budapest, 1976
YEARS IN SERVICE AT UT DALLAS:	7 years
RELATED EXPERIENCE:	Széchenyi Professor, Dept. of Telecommunications and Telematics, Technical University of Budapest, 1997 Director of Research of the High Speed Networks Laboratory, Technical University of Budapest, 1992-1997 Senior Research Associate, Dept. of Electrical and Computer Engineering, Boston University, 1996 Senior Associate Professor, Dept. of Telecommunications and Telematics, Technical University of Budapest, 1995 Senior Research Fellow, Dept. of Electrical and Computer Engineering, University of Massachusetts, 1991-1992 Associate Professor, Dept. of Telecommunications and Telematics, Technical University of Budapest,1982-1995 Assistant Professor, Dept. of Telecommunications and Telematics, Technical University of Budapest, 1976-1982
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I N THE LAST FIVE YEARS:	A. Farago, ``Algorithmic Challenges in Ad Hoc Networks", In: S. Basagni, M. Conti and S. Giordano and I. Stojmenovic (Eds.) *Mobile Ad Hoc Networking*, IEEE Press and Wiley, 2004. A. Farago and V.R. Syrotiuk, ``MERIT: A Scalable Approach for Protocol Assessment", Invited paper, *Mobile Networks and Applications (MONET), Spec. Issue on Mobile Ad Hoc Networks*, 8(2003), pp. 567-577. A. Farago, A. Szentesi and B. Szviatovszki, ``Inverse Optimization in High Speed Networks", *Discrete Applied Mathematics, Spec. Issue on Combinatorial and Algorithmic Aspects of Telecommunications*, 129(2003), pp. 83-98. A. Farago and V.R. Syrotiuk, ``Medium Access Control (MAC) Protocols", In: J. Proakis (Ed.), *Encyclopedia of Telecommunications*, Wiley 2002 A. Farago, A.D. Myers, V.R. Syrotiuk, and G. Zaruba. ``Meta-MAC Protocols: Automatic Combination of MAC Protocols to Optimize

	Performance for Unknown Conditions," *IEEE Journal on Selected Areas in Communications*, 18(2000), pp. 1670--1681. A. Magi, A. Szentesi, B. Szviatovszki, A. Farago, ``Dynamic Routing in ATM Networks", *Journal on Communications*, 50(1999) pp. 2-11. I. Chlamtac and A. Farago, ``A New approach to the Design and Analysis of Peer-to-Peer Mobile Networks", *Wireless Networks*, 5(1999/3), pp. 149-156.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Senior member, IEEE Member ACM Member, Janos Bolyai Mathematical Society, Hungary. Member, Scientific Society for Telecommunications, Hungary.
HONORS & AWARDS:	Niveau Award, Journal on Communications, 1988.
COURSES TAUGHT 2000-2005:	CS 4349 Advanced Data Structures and Algorithms CS 4390 Computer Networks CS 6385 Algorithmic Aspects of Telecommunication Networks
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Gopal Gupta
DATE OF BIRTH:	
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D, Computer Science, University of North Carolina at Chapel Hill (1992) M.S., Computer Science, University of North Carolina at Chapel Hill (1987) B.Tech., Computer Science, Indian Institute of Technology, Kanpur (May 1985)
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Research Associate in the research group of David H.D. Warren (of the Warren Abstract Machine fame, now retired) in the CS Department, University of Bristol, UK, 1989-1991 Faculty Member, Computer Science Department at New Mexico State University, 1992-2000**.**
CONSULTING, PATENTS, ETC.:	Knowledgesheet: A Spreadsheet Interface for Constraint Logic Programs for Tabular Problems. Patent Pending. (with S. Akhter). Contract with College of Arts & Sciences, NMSU, to design, implement and install logic based programming system for automated checking of graduation requirements (degree audit).(with Arthur Karshmer) Technology Transfer Agreement with ALS Inc., for building a parallel logic programming system based on the ALS (constraint) logic programming system. (with E. Pontelli, H-F. Guo, K. Villaverde)
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Qian Wang, G. Gupta, M. Leuschel. "Towards Provably Correct Code Generation via Horn Logical Continuation Semantics". In *Proc. International Conf. on Practical Aspects of Declarative Languages (PADL)* 2005. Springer Verlag. pp. 98-112. 2005. E. Pontelli, D. Ranjan, G. Gupta, B. Milligan. "Design and

Implementation of a Domain Specific Language for Phylogenetic Inference". *Journal of Bioinformatic and Computational Biology*, 1(2):2003. pp. 201-230.

G. Gupta, E. Pontelli, K. Ali, M. Carlsson, M. Hermenegildo, "Parallel Execution of Prolog Programs: A Survey". In *ACM Transactions on Programming Languages and Systems*, Vol. 23, No. 4, pp. 472-602.

G. Gupta. "Horn Logic Denotations and Their Applications. In The Logic Programming Paradigm: A 25 year perspective". Springer Verlag. pp. 127-160. April '98.

H-F Guo, B. Jayaraman, G. Gupta, M. Liu. "Optimization with Mode-Directed Preferences". In *ACM Conference on Principles and Practice of Declarative Programming*. 2005. To appear.

M. Nichols, Q. Wang, G. Gupta. "A VoiceXML-based Spoken Scripting Language for Voice-based Web Navigation". In *Human Computer Interaction Conference*, July 2005, Lawrence Erlbaum and Associates. To appear.

G. Gupta, S. Sunder Raman, M. Nichols. "DAWN: Dynamic Aural Web Navigation". In *Human Computer Interaction Conference*, July 2005, Lawrence Erlbaum and Associates. To appear.

R. Venkitaraman, G. Gupta. "Static Program Analysis of Embedded Executable Assembly Code". In *Proc. 7th International Conference on Compilers, Architectures, and Synthesis of Embedded Systems (CASES)*. ACM Press. 2004. pp. 157-164.

A. Karshmer, G. Gupta, K. Miesenberger, E. Pontelli, H. Guo, et al. "UMA: A System for Universal Mathematics Accessibility". In *Proc. ACM International Conference on Assistive Technology*. 2004. pp. 55-62.

SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:

Member, Association for Computing Machinery
- SIGPLAN
- SIGART

Executive Committee Member (2003-2007), Association for Logic Programming.

Co-founder and Co-coordinator, COMPULOG AMERICAS, a network of research groups in the Western Hemisphere engaged in research on Computational Logic.

Member of the Board (9/01-8/03), European Association for Programming Languages and Systems (EAPLS)

Conference Coordinator, Association for Logic Programming.

HONORS & AWARDS:

Member, Executive committee, Association of Logic Programming.
Member, Executive Committee, European Association for

Programming Languages and Systems.
Junior Faculty Enhancement Award in Computer Sciences, Oak Ridge Associated Universities, 1992.
Most Practical Paper Award, Sixth International Conference on Practical Aspects of Declarative Languages. 2004. (With H. F. Guo)
Best paper Award, Software Verification and validation Workshop, 2003. (With Q. Wang)
Recipient of Graduate School Fellowship, University of North Carolina, Chapel Hill.
Recipient of 'National Talent Search Scholarship', Government of India, 1981-1985.
Ranked sixteenth nationwide (among 100,000 students) in the Joint Entrance Examination of Indian Institute of Technology (IIT- JEE)

COURSES TAUGHT 2000-2003:	CS6371: Advanced Programming Languages
	CS7301: Programming Languages and S/W Engineering
	CS6374: Computational Logic
	CS6389: Formal Methods and Programming Methodology
OTHER ASSIGNED DUTIES:	Chair, ABET Accreditation (CS)
	Class scheduling
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Sanda M. Harabagiu
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Engineering, University of Southern California, 1997 PhD, Computer Science, University of Rome, Italy, 1994 Diploma Engineer in Computer Science and Electrical Engineering, Polytechnic Institute of Bucharest, Romania, 1983
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Assistant Professor, Computer Sciences, University of Texas, Austin, 2001-2002 Assistant Professor, Computer Science, Southern Methodist University, Austin, 1998-2001 Researcher, SRI International, 1997-1998
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Srini Narayanan and Sanda Harabagiu, "Question Answering Based on Semantic Structures" , in *Proceedings of the 20th International Conference on Computational Linguistics (COLING-2004)*, pp 693-701, 2004. Sanda Harabagiu, Steve Maiorano and Marius Pasca, "Open-Domain Textual Question Answering Techniques", in *Journal of Natural Language Engineering*, Vol. 9, No 3, September 2003, pp 3-44, Cambridge University Press, 2003. Dan Moldovan, Marius Pasca, Sanda Harabagiu and Mihai Surdeanu, "Performance Issues and Error Analysis in An Open-Domain Question Answering System", in *ACM Transactions on Information Systems*, Vol 21, No 2, pp 133-154, 2003. Mihai Surdeanu, Sanda Harabagiu, John Williams and Paul Aarseth, "The Using Predicate-Argument Structures for Information Extraction" , in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL-2003)*, pp 8-15, 2003.. Mihai Surdeanu, Dan Moldovan, and Sanda Harabagiu, "Performance Analysis of a Distributed Question Answering System", in *IEEE Transactions on Parallel and Distributed Systems*, Vol 13, No 6, pp 611-627, 2002. Sanda Harabagiu, Dan Moldovan and Joe Picone, "Open-Domain Voice-Activated Question Answering" , in *Proceedings of the 19th International Conference on Computational Linguistics*

(COLING-2002), pp 321-327, 2002.

Sanda Harabagiu, Steve Maiorano and Marius Pasca, "A Knowledge-Based Answer Engine for Open-Domain Questions" , in *International Journal of Artificial Intelligence Tools*, Vol 10, No 1-2, pp 99-224, World Scientific Publishing Company, 2001.

Marius Pasca and Sanda Harabagiu, "High Performance Question/Answering", in *Proceedings of the 24th Annual International ACL SIGIR Conference on Research and Development in Information Retrieval (SIGIR-2001)*

Sanda Harabagiu, Dan Moldovan, Marius Pasca, Rada Mihalcea, Mihai Surdeanu, Razvan Bunescu, Roxana Girju, Vasile Rus and Paul Morarescu, "The Role of Lexico-Semantic Feedback in Open-Domain Textual Question-Answering", in *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-2001)*.

SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE Computer Society AAAI ACL.
HONORS & AWARDS:	AQUAINT-II "AQUINAS: Answering Questions Using Inference and Advanced Semantics" AQUAINT-I "Computational Implicatures for Advanced Question Answering". ARP "Open-Domain Information Extraction". NSF CADRE "A Tool for Transforming WordNet into a Core Knowledge Base" NSF CAREER "Reference Resolution for Natural Language Understanding".
COURSES TAUGHT 2001-2004:	CS 6364 Artificial Intelligence CS 6322 Information Retrieval CS 6321 Discourse Processing
OTHER ASSIGNED DUTIES:	Director, Human Technology Research Institute Group Leader, Intelligence Systems Group 2004-2006 Member, Dean Search Committee, Jonsson School, UTD, 2002-2003 Member, Equipment Committee, Dept of Computer Science, UTD, 2002 Member, Dean of Engineering Search Committee, SMU, 1999-2000
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Co-Instructor with Dr. Srini Narayanan of the HLT-NAACL'2004 Tutorial on Semantic Inference for Question Answering. Co-Instructor with Prof. Dan Moldovan of the COLING'2002, IJCAI-2001, and NAACL'2001 Tutorial on Question Answering Systems.

NAME:	Herman W Harrison
DATE OF BIRTH:	July 25, 1943
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Physics, University of Texas, Austin, *1972* BS, Physics, University of Texas, Arlington, *1965*
YEARS IN SERVICE AT UT DALLAS:	4 years
RELATED EXPERIENCE:	Texas Instruments, Texas Tech University
CONSULTING, PATENTS, ETC.:	Patent: Algebraic Operating System (AOS) for Calculators
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	ACM, 1974 IEEE, 1975
HONORS & AWARDS:	
COURSES TAUGHT 2001-2004:	CS 1315: C++ and Java CS 2310/2110: Introduction to Digital Systems. CS2315: C++ and Java CS5330: Computer Science II.
OTHER ASSIGNED DUTIES:	ABET coordinator for CS2110/CS2310 Java Conversion committee
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Vasileios Hatzivassiloglou
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, Columbia University, 1998. M.Phil., Computer Science, Columbia University, 1994. MS, Computer Science, Columbia University, 1992. Diploma (five-year BS degree), Computer Science and Computer Engineering, University of Patras, Greece. 1990.
YEARS IN SERVICE AT UT DALLAS:	1 year
RELATED EXPERIENCE:	Research Scientist, Center for Computational Learning Systems, Columbia University. March 2004 – August 2004. Associate Research Scientist, Center for Computational Learning Systems, Columbia University. September 2004 - March 2004. Consultant, Department of Computer Science, Columbia University. Feb. 1998 – August 1998. Graduate Research Assistant, Department of Computer Science, Columbia University, September 1991 – Feb. 1998. Visiting Research Fellow, Information Sciences Institute, University of Southern California. May-August 1994, June-August 1996. Consultant, Bell Communications Research (Bellcore), October – Dec. 1995.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Hong Yu, Wong Kim, Vasileios Hatzivassiloglou and W. John Wilbur, “Unsupervised Approaches for Disambiguating Biomedical Abbreviations”, *ACM Transactions on Information Systems*, Jan 2004. Ivan Iossifov, Michael Krauthammer, Carol Friedman, Vasileios Hatzivassiloglou, Joel S. Bader, Kevin P. White and Andrey Rzhetsky, “Probabilistic Inference of Molecular Networks from Noisy Data Sources”, *Bioinformatics*, 20(8):1205-1212, Oxford University Press, 2004. Elena Filatova and Vasileios Hatzivassiloglou, “A Forma Model for Information Selection in Multi-Sentence Text Extraction”. *Proceedings of Twentieth International Conference on Computational Linguistics (COLING-04)*, August 2004. Hong Yu, Vasileios Hatzivassiloglou, Won Kim and W. John Wilbur. “Using MEDLINE as a Knowledge Source for Disambiguating Abbreviations in Full-Text Biomedical Journal Articles”. *Proceedings of the Seventeenth IEEE Symposium on Computer-Based Medical Systems (CBMS)*, June 2004.

SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Association for Computational Linguistics (ACL) North American Chapter of Association for Computational Linguistics(NAACL) International Society for Computational Biology(ISCB) American Association for Artificial Intelligence(AAAI) Association of Computing Machinery(ACM)
HONORS & AWARDS:	Excellence in Engineering Studies, Association of Greek Engineers, 1990. Excellence in Undergraduate Studies, Institution of State Academic Fellowships, Ministry of Education, Greece. 1986, 1987, 1988, 1989 and 1990. Second place in University entrance examination among nearly 30,000 persons applying to all Science and Engineering academic departments in Greece in 1985.
COURSES TAUGHT 2000-2005:	CS 7301: Natural Language Processing
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Program Committee Member and Reviewer, Annual International Conference on Intelligent Systems for Molecular Biology, 2005. Program Committee Member and Reviewer, International Conference on Computational Linguistics, 2004. Session Chair, Session on Lexical Semantics, Annual Meeting of the Association for Computational Linguistics, 2001.

NAME:	D. T. Huynh
DATE OF BIRTH:	
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D., Computer Science Department, University of Saarland, Germany, 1978 M.S., Computer Science Department, University of Saarland, Germany, 1977
YEARS IN SERVICE AT UT DALLAS:	19 Years
RELATED EXPERIENCE:	Professor and Department Head, University of Texas at Dallas. 1997-Date Professor Computer Science, University of Texas at Dallas. 1991-1997 Associate Professor, Computer Science, University of Texas at Dallas. 1986-91 Assistant Professor of Computer Science, Iowa State University. 1983-1986 Visiting Assistant Professor of Mathematics and Computer Science, University of Chicago. 1982-1983 Postdoctoral Research Associate, University of Saarland, Germany. 1978-1982 Teaching Assistant, University of Saarland, Germany. 1977-1978.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	"Connected D-Hop Dominating Sets in Ad Hoc Networks" (with T. Vuong. *Proc. Of the 6th World Multiconference on Systemics, Cybernetics and Informatics*, pp. 54-59, Florida, 2002. "Max-Min Cluster Formation in Wireless Ad Hoc Networks", (with A. Amis, R. Prakash & T. Vuong), *Proc. INFOCOM 2000*. "A Rearrangement Algorithm for Switching Networks Composed of Digital Symmetrical Matrices" (with Hai Nguyen), *Information Sciences,* Vol. 125, pp. 83-98, 2000. "Software Architecture Analysis: A Dynamic Slicing Approach" (with T. Kim, Y.-T. Song and L. Chung), *International Journal of Computer & Information Science*, Vol 1, no 2, pp. 91-103, 2000. "Adapting D-Hop Dominating Sets to Topology Changes in Ad Hoc Networks" (with T. Vuong) *Proc. Of ICCCN*, 2000.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A	Association of Computing Machinery (ACM) ACM SIGACT ACM SIGCOM IEEE Computer Society

MEMBER:	
HONORS & AWARDS:	
COURSES TAUGHT 2001-2004:	
OTHER ASSIGNED DUTIES:	CS Department Advisory Committee Graduate Admissions Committee. Faculty Search Committee TA Committee Graduate and Undergraduate Curriculum Committees
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Advisory Board Member, Journal of Automata, Languages and Combinatorics Chair, IEEE Symposium on Appl. Specific Software Engineering and Technology, 1998. Program Committee Member, International Conference on Computers, Science and Info., 2000. Program Committee Member, International Conference on Computation, Communication and Networks, 2000. **Reviewer:** National Science Foundation, Information and Computation, Information Processing Letters, Theoretical Computer Science, Hong Kong Research Grants Council, Natural Sciences and Engineering Research Council of Canada Information and Computation, Acta Informatica, SIAM Journal on Computing, Journal of Computer and System Science, Mathematical Systems Theory, International Journal of Foundations of Computer Sciences.

NAME:	Jason Jue
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Electrical & Computer Engineering, University of California at Davis, 1999. MS, Electrical Engineering, UCLA, 1991 BS, Electrical Engineering, University of California at Berkeley, 1990.
YEARS IN SERVICE AT UT DALLAS:	6 years
RELATED EXPERIENCE:	
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	Q. Zhang, V. Vokkarane, J. P. Jue, and B. Chen, "Absolute QoS Differentiation in Optical Burst-Switched Networks," to appear, *IEEE Journal on Selected Areas in Communications*, Nov. 2004. T. Zhang, K. Lu, and J. P. Jue, "Differentiated Contention Resolution for QoS in Photonic Packet-Switched Networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 22, no. 11, pp. 2523-2535, Nov. 2004. S. Yuan and J. P. Jue, "Dynamic Lightpath Protection in WDM Mesh Networks under Wavelength-Continuity and Risk-Disjoint Constraints," accepted for publication, *Computer Networks Journal (Elsevier)*. V. Vokkarane and J. P. Jue, ``Burst Segmentation: An Approach for Reducing Packet Loss in Optical Burst Switched Networks," *SPIE/Kluwer Optical Networks Magazine*, vol. 4, no. 6, pp. 81-89, November-December 2003. K. Lu, J. P. Jue, G. Xiao, and I. Chlamtac, ``Intermediate-Node Initiated Reservation (IIR): A New Signaling Scheme for Wavelength-Routed Networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 8, pp. 1285-1294, October 2003. V. Vokkarane and J. P. Jue, ``Prioritized Burst Segmentation and Composite Burst Assembly Techniques for QoS Support in Optical Burst Switched Networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 7, pp. 1198-1209, September 2003. S. Yuan and J. P. Jue, ``A Shared Protection Routing Algorithm for Optical Networks," *Optical Networks Magazine*, vol. 3, no. 3, pp. 32-39, May/June 2002. B. H. Simov, J. P. Jue, and S. Tridandapani, ``Integrating Security

	in the MAC Layer of WDM Optical Networks," *Photonic Network Communications*, vol. 4, no. 1, pp. 19-35, January 2002.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE
HONORS & AWARDS:	NSF Career Award, 2002
COURSES TAUGHT 2000-2003:	CS 6352: Performance of Computer Systems and Networks (Fall 2004) CS 6v81: Optical Networks (Spring 2004) CS 3305: Discrete Mathematics II (Fall 2004)
OTHER ASSIGNED DUTIES:	Faculty Search Committee, Telecommunications Engineering Program Bylaws Committee, Computer Science Dept. Faculty Secretary, Computer Science Dept. Networking Group Representative, Computer Science Dept.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Shyam S. Karrah
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	BS (Hons) MS (Computer Science)
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Professor of Computer Science, Department Chair (1998-2000), New Mexico Junior College, Hobbs, New Mexico, 1994-2000
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	National Academic Advising Association
HONORS & AWARDS:	
COURSES TAUGHT 2000-2003:	Organization of programming Languages
OTHER ASSIGNED DUTIES:	Planning and coordinating the Graduate Orientation and Transfer & Waiver seminars. Advising graduate students on academic issues: admission, curriculum and graduation requirements, academic policies, and academic standards for probation.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Latifur Khan
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of Southern California**,** 2000 MS, Computer Science, University of Southern California, 1996 BSc, Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh, 1993
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	L. Khan and D. McLeod, “Audio Structuring and Personalized Retrieval Using Ontologies,” in *Proc. of ACM/IEEE Advances in Digital Libraries*, Library of Congress, Washington, DC, pp. 116-126, May 2000. L. Khan, D. McLeod and E. Hovy, “Retrieval Effectiveness of Ontology-based Model for Information Selection,” the *VLDB Journal: The International Journal on Very Large Databases*, ACM/Springer-Verlag Publishing, Vol. 13(1): 71-85 (2004). F. Luo, L. Khan , F. Bastani, I-Ling Yen and J. Zhou, “A Dynamical Growing Self-Organizing Tree (DGSOT) for Hierarchical Clustering Gene Expression Profiles,” the *Bioinformatics Journal*, Oxford University Press, UK, 20(16): 2605-2617 (2004). L. Khan, D. McLeod and E. Hovy, “A Framework for Effective Annotation of Information from Closed Captions Using Ontologies," to appear in Journal of Intelligent Information Systems, Kluwer Publisher. M. Gupta, M. Tu, L. Khan, F. Bastani, and I-Ling Yen, "A study of Algorithms for Handling Location Dependent Continuous Queries in the Mobile Environment," to appear in the *Knowledge and Information Systems Journal (KAIS)*, Publisher: Springer-Verlag London Ltd.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	ACM IEEE
HONORS & AWARDS:	

COURSES TAUGHT 2001-2004:	CS 6360 Database Design Data Mining CS 6V81 Data Management for Mobile Computing CS 6V81 Distributed Multimedia Information Management
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Director of DBL@UTD, UTD Database Laboratory

NAME:	Lawrence A. King
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	MS in Computer Science, New Mexico State University, Las Cruces, NM MBA, California State University, Northridge, California BS in Aeronautics and Astronautics, M.I.T, Cambridge, MA
YEARS IN SERVICE AT UT DALLAS:	4 years
RELATED EXPERIENCE:	Senior Research Scientist, Honeywell Laboratories, Minneapolis, MN Research Scientist, Motorola Artificial Intelligence Laboratory, Scottsdale, AZ Aeronautical Engineer, USAF (Active Duty), Wright-Patterson AFB, Ohio Aeronautical Engineer, Lockheed Missiles and Space Co., Sunnyvale, CA
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	Sun Certified Programmer for the Java 2 Platform
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	
COURSES TAUGHT 2000-2003:	CS 1315 Computer Science I CS 2315 Computer Science II CS 3336 Programming in Java
OTHER ASSIGNED DUTIES:	**UTD administrative duties** Undergraduate Faculty Advisor ABET course coordinator for CS2315, Computer Science II Selected and supervised Special Tutor for C students entering CSII Participated in CS/SE Faculty Advisor Information Seminar for undergraduate CS and SE majors **School committees** Chair, Java Curriculum Conversion Committee, Department of Computer Science. During Fall 2003, CS1315 was successfully

converted from C to Java. During Spring 2004, CS2315 was converted from C to Java.

Chair, Curriculum Conversion Committee, Department of Computer Science: Created catalog description for CS1301, Programming Fundamentals, and revised descriptions for CS1315, CS2315, CS3345, and CS4349 for integrated content

Member of the Curriculum Committee, Department of Computer Science

Member of the Industrial Advisory Board, Department of Computer Science

Special service contributions

1. Design and implementation of a TA scheduling application
2. Performed TA scheduling for Computer Science Department, Spring/Fall 2003 and 2004,

SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:

NAME:	Radha Krishnan
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD (Mechanical Engineering) Penn State M.S. (Mechanical Engineering) Penn State B. Tech. (Mechanical Engineering) Indian Inst of Tech
YEARS IN SERVICE AT UT DALLAS:	4 years
RELATED EXPERIENCE:	Advisor, Future Technologies, Nortel Networks, Sep 2000 – Apr 2001. Senior Member of Scientific Staff, Nortel Networks, Nov 1999 – Aug 2000 Consultant, Ericsson Inc, May 1998 -- July 1999. Visiting Professor, University of Texas at Dallas, Jun 1995 -- Aug 1999. Asst. Professor, Univ. of Maryland, Jan 1992 -- Aug 1997 Consultant, Penn. Transportation Institute, EDI, ARP, Summers 1987 – 1991 Research Assistant, The Pennsylvania State Univ., Jan 1987 – Dec 1991
CONSULTING, PATENTS, ETC.:	A Mobile Marketplace Architecture for a Next Generation Network, supporting M-Commerce", patent filed.
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Balaji Raghavachari, Gopal Gupta and Radha Krishnan, "Database design for hazardous materials", in preparation. Radha Krishnan, Balaji Raghavachari, Gopal Gupta, Jim Staves and Doug Harris, "An Emergency Planning System for the 21st Century: Case Study of E-Plan", in preparation. Radha Krishnan, "Towards a more logical organization of network elements and their functions: the case for rethinking the Internet as a content-dispensing mechanism", in preparation. Radha Krishnan, "Performance Evaluation of an Advanced Wireless Internet", awaiting clearance. Mike Hall, Radha Krishnan, Narendra Kanar and Anu Appaji, "A Mobile Marketplace Architecture for a Next Generation Network supporting M-Commerce", submitted. Radha Krishnan, Balaji Raghavachari, " The Directed Minimum Degree Spanning Tree Problem", to be presented at the *Foundations of Software Technology and Theoretical Computer Science*, Dec 13-15, Bangalore, India. Imrich Chlamtac, Radha Krishnan, C. Petrioli and J. Redi,

	"Energy conservation in access protocols for mobile computing and communications", to appear in the special issue on *Personal Digital Assistants of the J. Microprocessors and Microsystems*. Radha Krishnan, et. al., "A Framework for the Performance Evaluation of Integrated Networks", in *Proc. Wireless Communications and Networking Conference*, Sept. 23 - 28, 2000, Chicago, IL.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	VoIP Performance Characterization Project (Summer and Fall 1998) for Ericsson. Won the best project award in the company, among all its world-wide units. Mentioned in People in Computer Vision, URL: http://www.cs.hmc.edu/~eck/computer-vision-handbook/vision-people.html Invited to Chair a session in ISATA Conference, Aachen, Germany, October 1994. Invited Talk at the Johns Hopkins University, November 1993. Invited Talk at the University of Rochester, September 1993. Invited Talk at the Johnson Medical Center, UPenn, August 1993. Invited participant, "Reasoning about Functionality" AAAI Workshop, July 1993. Invited to Chair the "Inspection, Mapping, Coding and Monitoring" session of the IEEE Computer Vision and Pattern Recognition Conference, June 1993. Invited Talk at the Center for Automation Research, College Park, August 1991. Invited Talk at NASA Lewis Research Center, May 1991.
COURSES TAUGHT 2000-2003:	CS 2305 Discrete Mathematics for Computing I CS 3305 Discrete Mathematics for Computing II CS 3333 Data Structures CS 4347 Database Systems CS 4349 Advanced Algorithm Design and Analysis CS 5333 Discrete Structures CS 6387 Computer-Aided Software Engineering
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Rafael Lacambra
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	MS, Computer Science (Major In Computer Graphics And Multimedia), George Washington University, 1997 BS, Computer Engineering, Universidad Nacional Autonoma De Mexico, Mexico City, Mexico, 1991
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Silicon Graphics, Southern Education Center, Dallas, Texas. 1999-2001 Education Specialist - Computer Graphics / Networking / IRIX Operating System Provided training expertise (OpenGL programming, Network Administration, System Administration and Advanced System Administration that enabled the company to cover a broad curriculum with fewer instructors. Consistently achieved ratings of 9.8/10 or higher as instructor. Designed and developed custom system administration and graphics courses based on assessment of client skills and needs. Lead instructor/developer/reviewer for Advanced Network Administration. Developer/reviewer for Network Administration and OpenGL. Cactun Espacio Digital, Mexico City/D.F., Mexico, 1997-1999 Authorized trainer for SGI and SUN Mexico in the UNIX operating system and TCP/IP networking. Professor of "Graphical User Interfaces" at Universidad Nacional Autonoma de Mexico. Universidad Nacional Autonoma De Mexico (UNAM), 1997-1999 Researcher in Computer Graphics, Applied Computing Department Formed and led a team to develop and promote the use of virtual reality in archaeological sites. Performed research and wrote two papers on "Virtual Reality and Animation, two alternative tools for the visualization of archaeological sites." Organized the creation of the Alias/Wavefront training room at UNAM and obtained cost-free courses and certification for 3 trainers. Universidad Nacional Autonoma De Mexico, 1991-1995 Head of Visualization Laboratory, Computing for Research Created the Visualization and Networking areas of the "Plan de Becarios en Supercomputo" scholarship courses in supercomputing topics, which trained undergraduate and graduate students to act as research support personnel

CONSULTING, PATENTS, ETC.:	Cactun Espacio Digital, Mexico City/D.F., Mexico 1997 - 1999 Independent Consultant, Computer Graphics / Networking / UNIX Founded one of the first consulting firms in Mexico specializing in web topics. Established service standards that earned bonuses for early delivery of the finished products
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	Two grants by Cray Research for UNAM to be part of the Khoros Consortium One grant awarded by Cray Research to pay for AVS (graphical simulation) licenses for the Visualization lab. One grant awarded by Cray Research to start a joint project between Jefferson Hospital in Philadelphia and the National University of Mexico (UNAM) which main topic was the study of cerebral aneurysms
COURSES TAUGHT 2000-2003:	CS3375 Principles of UNIX CS5375 Principles of UNIX (Graduate) CS2315 Computer Science 2 (Using Java)
OTHER ASSIGNED DUTIES:	Undergraduate Faculty Advisor Computer Science/ Software Engineering
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Joseph A. Leubitz
DATE OF BIRTH:	April 21, 1945
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	MS, Computer & Information Science, Ohio State University, Columbus, OH, 1971. BSEc, Accounting, Wharton School of Finance & Commerce, University of Pennsylvania, Philadelphia, PA, 1967
YEARS IN SERVICE AT UT DALLAS:	4 years
RELATED EXPERIENCE:	Berkeley College, Woodridge, NJ, Instructor (1993-1994) Collin County Community College, Plano, TX, Instructor (1991) Accelerated Computer Training, Northbrook, IL, Owner (1984-1985) Lake Forest School Of Management, Lake Forest, IL, Adjunct Instructor (1984) Northwestern University, Evanston, IL Instructor (1976-1978) Kendall College, Evanston, IL, Instructor (1974-1975)
CONSULTING, PATENTS, ETC.:	35 years consulting experience (IT audits, security, system design/implementation, project management)
STATE(S) IN WHICH REGISTERED:	CPA with CITP specialization. (Texas, Indiana, and Illinois) CCP-Management. CISA. PMP
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Association for Computing Machinery American Institute of CPAs Project Management Institute Information Systems Audit & Control Association
HONORS & AWARDS:	
COURSES TAUGHT 2001-2004:	CS/SE 3354 Software Engineering CS 3385 Ethics, Law, Society, and Computing CS/SE 4347 Database Systems SE 4367 Software Testing, Validation, Verification and Quality Assurance SE 4381 Software Project Management CS 6388 Software Project Management
OTHER ASSIGNED DUTIES:	Faculty Advisor

SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:

NAME	Rym Mili
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of Ottawa, Canada Doctorate de Spécialité, Computer Science, University of Tunis, Tunisia. Engineering Degree, Computer Science, University of Tunis, Tunisia.
YEARS IN SERVICE AT UT DALLAS:	10 years
RELATED EXPERIENCE:	Assistant Professor, School of Engineering and Computer Science, University of Texas at Dallas, 1995-2002. Lecturer, Institute for Government Informatics Professionals, Ottawa, Canada. 1993-1995 Lecturer, Department of Computer Science, University of Ottawa, Canada, 1992-1994
CONSULTING, PATENTS, ETC.:	Patent pending. R. Mili and S. Zalila. An Effective Data Entry Method. The invention is a prime candidate for all applications where we want to spare the space of traditional keyboards. Consulting, Philips, Sussex, England.
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	R. Castello and R. Mili, Visualizing Graphical and Textual Formalisms, *Information Systems*, vol. 28, pp. 753-768,Elsevier, 2003. R. Castello, R. Mili and I. G. Tollis, Visualizing Statecharts with ViSta, book chapter, *Graph Drawing Software: Mathematics and Visualization*, P. Mutzel and M. Juenger (eds.), pp. 299-319, Springer Verlag, 2003. R. Castello, R. Mili and I. G. Tollis, ViSta: A Tool Suite for the Visualization of Behavioral Requirements, *Journal of Systems and Software*, Elsevier, vol. 62, pp141-159, 2002. R. Castello, R. Mili and I. G. Tollis, Automatic Layout of Statecharts, *Software Practice and Experience*, vol. 32, pp. 25-55, John Wiley, 2002.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Member, IEEE Computer Society. Member, Association of Computing Machinery.
HONORS & AWARDS:	*NSF*, Using NLP Tools for Requirements Visualization. September 2001-September 2003. Sandia National Laboratories, Visualizing Software Requirements, with I. G. Tollis, November 1998-August 1999.
COURSES TAUGHT 2000-2005:	Software Engineering (graduate and undergraduate). Project Planning and Management (graduate), Object Oriented Analysis and Design (graduate) Software Maintenance and Re-engineering (graduate) Software Reuse (graduate),

Data Structures (graduate)

OTHER ASSIGNED DUTIES:	Member, *Graduate Curriculum committee* 1997-2004 Member, *Ph.D. Qualifying Exam committee* 2002-2004 Member, *Masters-Research Track committee* 2003-2004 Member, *Admission committee* 1997-2001 Member, *Search committee* 2002-2003 Member, *ABET accreditation committee* 2002-2003
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Member, *University Committee on Effective Teaching* 2001-2003 Member, *ECS Committee on Effective Teaching* 2004-2005

NAME:	Neeraj Mittal
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D. in Computer Science, The University of Texas at Austin, 2002 M.S. in Computer Science, The University of Texas at Austin, 1997 B.Tech. in Computer Science and Engineering, Indian Institute of Technology, Delhi, 1995
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Graduate Research Assistant, Department of Electrical and Computer Engineering, The University of Texas at Austin, 1999-2002
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	Neeraj Mittal and Vijay K. Garg, “Techniques and Applications of Computation Slicing,” Accepted for publication in *Distributed Computing (DC).* Neeraj Mittal and Vijay K. Garg, “Finding Missing Synchronization in a Distributed Computation using Controlled Re-execution,” *Distributed Computing (DC),* Online First, 2004. Neeraj Mittal, S. Venkatesan and Sathya Peri, “Message-Optimal and Latency-Optimal Termination Detection Algorithms for Arbitrary Topologies,” In *Proceedings of the 18th International Symposium on Distributed Computing (DISC)*, 2004. Sathya Peri and Neeraj Mittal, “On Termination Detection in an Asynchronous Distributed System,” *In Proceedings of the 17th ISCA International Conference on Parallel and Distributed Computing Systems (PDCS)*, 2004. Neeraj Mittal, Alper Sen, Vijay K. Garg and Ranganath Atreya, “Finding Satisfying Global States: One for All and All for One,” *In Proceedings of the 18th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2004. Ranganath Atreya, Neeraj Mittal and Vijay K. Garg. “Detecting Locally Stable Predicates without Modifying Application Messages,” In *Proceedings of the International Conference on Principles of Distributed Systems (OPODIS)*, 2003. Neeraj Mittal and Vijay K. Garg , “Software Fault Tolerance of Distributed Programs using Computation Slicing,” In *Proceedings of the 23rd IEEE International Conference on Distributed*

	Computing System (ICDCS), 2003. Neeraj Mittal and Vijay K. Garg, “Computation Slicing: Techniques and Theory”, *In Proceedings of the 15th International Symposium on Distributed Computing (DISC)*, 2001. Neeraj Mittal and Vijay K. Garg, “On Slicing a Distributed Computation,” In *Proceedings of the 21st IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2001. Neeraj Mittal and Vijay K. Garg, “On Detecting Global Predicates in Distributed Computations,” In *Proceedings of the 21st IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2001.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Association for Computing Machinery (ACM) IEEE (Institute of Electrical and Electronics Engineers) Computer Society
HONORS & AWARDS:	MCD Graduate Fellowship, University of Texas at Austin, 1995-1997 Suresh Chandra Memorial Award, Indian Institute of Technology, Delhi, 1995 Certificate of Merit, Indian Institute of Technology, Delhi, 1991-1994 Gold Medal, Mathematics Examination, Ramanujan Society for Born Mathematicians, New Delhi, India, 1991 Junior Science Talent Search Scholarship, Directorate of Education, Delhi, India, 1987-1989
COURSES TAUGHT 2000-2003:	CS 6378: Advanced Operating Systems CS 6380: Distributed Computing CS 7301: Advances in Distributed Computing
OTHER ASSIGNED DUTIES:	Chair, Advanced Operating Systems Qualifying Examination Committee, 2004 Member, Equipment Committee, Department of Computer Science, 2003-2004
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Dan I. Moldovan
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Electrical Engineering and Computer Science from the Columbia University, 1978 MS, Electrical Engineering and Computer Science from the Columbia University, 1974 BS, Electrical Engineering, Polytechnic Institute of Bucharest, 1969
YEARS IN SERVICE AT UT DALLAS:	4 years
RELATED EXPERIENCE:	**A. Research Policy at National Level** Program Director, National Science Foundation, Washington D.C. (Sabbatical year 1987 - 1988) Directed Experimental Systems Program in the Division of Microelectronics and Information Processing Systems **B. Academic** Professor of Computer Science, University of Texas at Dallas 8/2001- Present Chairman of the Computer Science and Engineering Department Southern Methodist University, Dallas. 8/1994- 7/1998 Professor of Computer Science and Engineering, and Director of the Parallel and Distributed Computer Systems Laboratory Southern Methodist University, Dallas, Texas 75275 8/1993- 8/2001 Associate Professor of Computer Engineering, and Director of the Parallel Knowledge Processing Laboratory, University of Southern California, Los Angeles. 5/1986- 8/1993 Assistant Professor of Computer Engineering, University of Southern California, Los Angeles. 9/1981- 5/1986 Assistant Professor of Electrical Engineering, Colorado State University, Fort Collins, Colorado. 9/1979- 8/1981
CONSULTING:	**Industry:** Technical consultant to Aerojet ElectroSystems, Hughes Research Laboratory, TRW. 1976-1979 **Member of Technical Staff - Bell Laboratories, Holmdel, New Jersey**
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Dan I. Moldovan, "*Parallel Processing: From Applications to Systems*," Morgan Kaufmann Publishers, 1993, San Mateo, California, (567 pages). "On the Role of Information Retrieval and Information Extraction in Question Answering Systems" in *Information Extraction in the Web Era*, Springer, 2003, 129-147. "Textual Question Answering", in *Handbook of Natural Language Processing*, Oxford Press 2003, R. Mitkov, editor, 560-582.

Dan Moldovan et al., "Performance Issues and Error Analysis in an Open-Domain Question Answering System", *ACM Transactions on Information Systems*, vol 21, nr 2, pp 133-154.
Vasile Rus, Dan I. Moldovan, "High Performance Logic Form Transformation" *International Journal on Artificial Intelligence Tools* 11(3): 437-454, 2002.
Mihai Surdeanu, Dan I Moldovan and Sanda Harabagiu, "Performance Analysis of a Distributed Question Answering System", *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 6, pp 579 - 596, June 2002.
Mihai Surdeanu and Dan I Moldovan, "Design and Performance of a Distributed Java Virtual Machine", *IEEE Transactions on Parallel and Distributed Systems*, vol 13, no. 6, pp 611-627, June 2002.
Dan Moldovan, Roxana Girju and Adriana Badulescu, "Learning Semantic Constraints for the Automatic Discovery of Part-Whole Relations", in *Proceedings of the HLT/NAACL 2003 Conference*, May 2003, Edmonton, Canada.

PROFESSIONAL MEMBERSHIPS:

Member of ACM, AAAI, ACL, IEEE Senior Member

COURSES TAUGHT 2000-2003:

CS 6320, Natural Language Processing
CS 6375, Machine Learning

RESEARCH:

Current Research:
Direct the InterVoice Bright Research Center in the Human Language Technology Research Institute at UTD.
Co-direct the development of a state-of-the-art Question Answering system.
Current Research Grants:
1. InterVoice Bright, 2002-2006, Research in Automatic Speech Recognition Systems, ($1,000,000).
2. ATP State of Texas, 2002-2004, Text Mining for Telecommunications ($240,000)
3. NSF, 2000-2005, to develop a Tool for automatic transformation of WordNet into a Knowledge Base ($700, 000).

SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:

Services to professional organizations:
Vice Chair for the NSF Workshop on High Performance Computing and Communication: Vision, Natural Language and Speech Processing, and Artificial Intelligence, February 1992.
Member of several NSF Panels for Presidential Young Investigators, Research Initiation, and CAREER Awards.
Area Editor for International Journal of Mini and Microcomputers.
Area Editor for Journal of Parallel and Distributed Computing.
Co-chairman for The First International Workshop on Parallel Processing for Artificial Intelligence; organized in conjunction with IJCAI-91, Sydney, Australia.
Served as a reviewer for books and journal papers in the area of digital systems and parallel processing.
Conference session organizer and chairman.

NAME:	Yu-Chung (Vincent) Ng
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D. in Computer Science, Cornell University, 2004 M.S. in Computer Science, Carnegie Mellon University, 2002 B.S. in Computer Science, Carnegie Mellon University, 1997
YEARS IN SERVICE AT UT DALLAS:	1 year
RELATED EXPERIENCE:	Research Assistant, Cornell University, 1999-2004. Teaching Assistant, Cornell University, 1998-99.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	"Detecting Discrepancies in Numerical Estimates Using Multi-document Hypertext Summaries" Michael White, Claire Cardie, Vincent Ng, and Daryl McCullough. *Proceedings of the Second International Conference on Human Language Technology Research* (HLT-02), 2002. "Multi-document Summarization via Information Extraction". Michael White, Tanya Korelsky, Claire Cardie, Vincent Ng, David Pierce, and Kiri Wagstaff. *Proceedings of the First International Conference on Human Language Technology Research* (HLT-01), 2001. "Detecting Discrepancies and Improving Intelligibility: Two Preliminary Evaluations of RIPTIDES". Michael White, Claire Cardie, Vincent Ng, Kiri Wagstaff, and Daryl McCullough. *Proceedings of the 2001 Document Understanding Conference* (DUC-01), 2001. "Learning Noun Phrase Anaphoricity to Improve Coreference Resolution: Issues in Representation and Optimization". Vincent Ng. *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (ACL-04), 2004. "Improving Machine Learning Approaches to Coreference Resolution. Vincent Ng and Claire Cardie". *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-02)*, 2002. "Identifying Anaphoric and Non-Anaphoric Noun Phrases to Improve Coreference Resolution". Vincent Ng and Claire Cardie. *Proceedings of the 19th International Conference on Computational Linguistics (COLING-02)*, 2002. "Combining Sample Selection and Error-Driven Pruning for

	Machine Learning of Coreference Rules". Vincent Ng and Claire Cardie. *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-02)*, 2002. "Examining the Role of Statistical and Linguistic Knowledge Sources in a General-Knowledge Question Answering System". Claire Cardie, Vincent Ng, David Pierce, and Chris Buckley. *Proceedings of the Sixth Applied Natural Language Processing Conference* (ANLP-2000), 2000. "Weakly Supervised Natural Language Learning Without Redundant Views. Vincent Ng and Claire Cardie". *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, 2003. "Bootstrapping Coreference Classifiers with Multiple Machine Learning Algorithms". Vincent Ng and Claire Cardie. *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP-03)*, 2003.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	American Association for Artificial Intelligence (AAAI) Association of Computational Linguistics (ACL) ACL Special Interest Group for Linguistic Data and Corpus-Based Approaches to NLP(SIGDAT)
HONORS & AWARDS:	Member of Phi Beta Kappa and Phi Kappa Phi Carnegie Melon University Dean's List: Fall 94, 95; Spring 95, 96, 97 Recipient of EDS Scholarship, Fall 1996.
COURSES TAUGHT 2000-2004:	CS 2305 Discrete Mathematics for Computing I
OTHER ASSIGNED DUTIES:	Computer Science Graduate Admissions Committee, 2005.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Program Committee Member, 43rd Annual Meeting of the Association for Computational Linguistics (ACL), 2005. Program Committee Member, Nineteenth International Joint Conference on Artificial Intelligence (IJCAI), 2005.

NAME:	Simeon Ntafos
DATE OF BIRTH:	October 23, 1952
ACADEMIC RANK:	Associate Dean, Erik Jonsson School of Engineering & Computer Science Professor and Associate Department Chair
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	B.S. in Electrical Engineering, Wilkes College, 1974; M.S. in Electrical Engineering, Northwestern, 1977 Ph.D. in Computer Science, Northwestern, 1979
YEARS IN SERVICE AT UT DALLAS:	26 years
RELATED EXPERIENCE:	1978-1979 Visiting Assistant Professor, Northwestern 1979-1984 Assistant Professor, Computer Science, UTD 1984-1994 Associate Professor, Computer Science, UTD 1985-1987 Program Head, Computer Science, UTD 1994- present Professor, Computer Science, UT-Dallas. 1998 -2004 Associate Department Chair, CS, UTD
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	“Testing and the Cost of Field Failures”, ISSRE-99 Fast Abstracts, Nov. 1999, pp. 13-14. “Improved Testing Using Failure Cost and Intensity Profiles”, Proc. ASSET-2000, pp. 126-130, March 2000 (with V. Benson). "On Comparisons of Random, Partition, and Proportional Partition Testing", IEEE Transactions on Software Engineering, Vol. 27, No. 10, pp. 949-960, Oct. 2001. “A Geometric Approach for finding HPD-Credible Sets with Applications”, Applied Mathematics for Computation, Vol. 125, pp. 195-207, 2002 (with L. Gewali and A. Singh).
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	ACM, ACM SIGACT, ACM SIGSOFT
HONORS & AWARDS:	
COURSES TAUGHT 2001-2004:	CS 6367 Software Testing, Validation and Verification Spring 2003 CS 6367 Fall 2003 CS 6367 Spring 2004 CS 6367
OTHER ASSIGNED	ABET Coordinator for BS-CS (3/04) and BS-SE (5/92);

DUTIES:	Faculty Senate (1997-); Academic Council (2003-) Secretary of the faculty (2004-) Committee on Qualifications of Academic Personnel (2002-vice Chair; 2001-member) PhD Committee – CS (Chair 2001-2003; Ex-officio 2003-) 2001-02 CS Search Committee 2000-01 CS Search Committee (Chair) 1999-01 Admissions Committee (Chair) 1999-01 EE&CS Personnel Review Committee (elected)
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	2005 UTA Advising Conference Panelist: International Test Conference, 1992. Program Committee Chair – ASSET 1998 Program Committee – COMPSAC 1999 General Chair – ASSET 2000

NAME:	Greg Ozbirn
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	MS degree in Computer Science from Southern Methodist University in 2000. BS degree in Computer Science from Harding University in 1987
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	2000-2001 Computer Science professor at Collin County Community College 1988-2000 Programmer/Analyst at Texas Instruments, Dallas. 2000-2003 Adjunct Computer Science teacher at SMU 1994-2000 Adjunct Computer Science teacher at Richland College 1993-2000 Adjunct Computer Science teacher at Collin County Community College
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	Microsoft Certified Solution Developer
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	Excellence in Teaching Award, University of Texas at Dallas, Spring 2003
COURSES TAUGHT 2001-2004:	CS 2315 Computer Science II CS 3345 Data Structures CS 3354 Software Engineering CS 4348 Operating Systems CS 6354 Advanced Software Engineering
OTHER ASSIGNED DUTIES:	ABET Coordinator for CS 4348 Operating Systems. Member of team to convert curriculum to Java.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Volunteer in effort to become ABET accredited.

NAME:	Ivor Page
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D in Computer Science, Brunel University, UK. 1979. BSc (Honors) in EE, Brunel University, UK. 1968
YEARS IN SERVICE AT UT DALLAS:	24 years
RELATED EXPERIENCE:	Associate Dean of Undergraduate Education, ECS at UTD, 1998-2004 Director of Collegiums Five Honors Program UTD, 1998-1998 Associate Program head Computer Science UTD, 1997-1998 Interim Program Head, Computer Science UTD, 1995-1997 Associate Professor in Computer Science UTD, 1987-date Assistant professor in Computer Science, 1981-1987 Lecturer in Computer Science with tenure, Brunel University, 1971-1981 Research Engineer for Elliott Automation Radar Systems, UK, 1970-1971 Research and Designing Engineer, ICL R & D Lab, UK, 1968-1970
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	ACM, ACM SIGGRAPH TSTA Texas State Teachers Association TFA Texas Faculty Association
HONORS & AWARDS:	
COURSES TAUGHT 2001-2004:	CE 6305 Computer Arithmetic CS 6386 Telecom's SW Design CS/CE/EE 6304 Computer Architecture CE 6305 Computer Arithmetic CS 6386 Telecom's SW Design CE/CS/EE 6304 Computer Architecture CS 4340 Computer Architecture
OTHER ASSIGNED	Ph.D Committees: Mei Yang Dr. Zheng's student

DUTIES:	Don Montgomery. Dr. Raghavachari's Student Member of ECS School Committee on teaching effectiveness Member of CS Curriculum Committee Ex Officio member of the Academic Affairs Committee charges with writing the first set of bylaws for the Jonsson School Member of the Faculty Senate Committee on Faculty Standing and Conduct
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Coach for student teams competing in the ACM Intercollegiate student programming contests Co-Chair of the Texas Higher Education Coordinating Board Advisory Committee for the Field of Study in Computer Science

NAME:	Balakrishnan Prabhakaran
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science & Engineering, Indian Institute of Technology, Madras, India, 1995. MS, Computer Science & Engineering, Indian Institute of Technology, Madras, India, 1990. BEng, Electronics & Communications, Madurai-Kamaraj University, 1986.
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Assistant Professor, School of Computing, National University of Singapore, 1997-2001 Visiting Research Faculty, Computer Science, University of Maryland, 1995-1997. Scientific Officer, Computer Science & Engineering, Indian Institute of Technology, Madras, India, 1989-1996.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	B. Prabhakaran, Multimedia Database Management Systems, Kulwer Academic Publishers, Boston. B. Prabhakaran, Multimedia Synchronization, Chapter 6, Multimedia Systems and Techniques, Editor: Prof. Borko Furht, Kluwer Academic Publishers, Boston. B. Prabhakaran,, H. Zhu, M. Li, and I. Chlamtac, „Survey of quality of ervice in IEEE 802.11 networks," *Mobility and Resource Management/IEEE Wireless Communications magazine, 2004.* S. Emilda, L. Jacob, O. Daescu, and B. Prabhakaran,, "Flexible disk scheduling strategies for multimedia presentation servers," *Multimedia Tools and Applications*, accepted for publication. E. Hwang and B. Prabhakaran,, "Application-layer protocol for collaborative multimedia presentations," *Multimedia Tools and Applications*, 2003. E. Hwang and B. Prabhakaran,, "Unified read requests," *Multimedia Tools and Applications*, 2003.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A	

MEMBER:	
HONORS & AWARDS:	NSF Career Grant, 2003
COURSES TAUGHT 2000-2003:	CS 6378 Advanced Operating Systems
OTHER ASSIGNED DUTIES:	Graduate Admissions Committee, Dept. of Computer Science, UTD, 2001-date. Teaching Assistants Committee, Dept. of Computer Science, UTD, 2003-date. Undergraduate Program Committee, Telecom Engineering Program, UTD.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Ravi Prakash
DATE OF BIRTH:	December 24, 1968
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D, Computer and Information Science, The Ohio State University, 1996 MS, Computer and Information Science, The Ohio State University, 1991 BTech, Computer Science & Engineering, Indian Institute of Technology, Delhi, 1990.
YEARS IN SERVICE AT UT DALLAS:	8 years
RELATED EXPERIENCE:	Visiting Assistant Professor, Computer Science, University of Rochester, 1996-1997 Presidential Fellow, Computer and Information Science, The Ohio State University, 1996 Teaching & Research Assistant, Computer and Information Science, The Ohio State University, 1990-1995.
CONSULTING, PATENTS, ETC.:	Williams-Pyro, Inc. (consulting)
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	M. Thoppian and R. Prakash. "A Distributed Protocol for Dynamic Address Assignment in Mobile Ad Hoc Networks." To appear in the IEEE Transactions on Mobile Computing. S. Nesargi and R. Prakash. "Distributed Wireless Channel Allocation in Networks with Mobile Base Stations," *IEEE Transactions on Vehicular Technology*, 2002. S.R. Gandham, M. Dawande, R. Prakash and S. Venkatesan. "Energy-Efficient Schemes for Wireless Sensor Networks with Multiple Mobile Base Stations." Proceedings of IEEE Globecom, December 2003. R. Prakash. "A Routing Algorithm for Wireless Ad Hoc Networks with Unidirectional Links," *ACM/Baltzer Wireless Networks Journal*, 2001. R. Prakash, Z. Haas, and M. Singhal. "Load-Balanced Location Management for Mobile Systems using Quorums and Dynamic Hashing," *ACM/Baltzer Wireless Networks (WINET) Journal*, 2001. K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash. "A Feedback Based Scheme for Improving TCP Performance in Ad Hoc Networks," 2001. R. Prakash and R. Baldoni. "Causality and Spatial-Temporal

	Ordering of Events in Mobile Systems," To appear in *ACM Baltzer Journal on Mobile Networks and Applications (MONET).*
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE
HONORS & AWARDS:	Career Award, NSF, 2001-2006 Excellence in Teaching Award, 1999-2000,2001-2002, ECS, UTD Presidential Fellowship, Ohio State University, 1996
COURSES TAUGHT 2000-2004:	CS 3345, Algorithm Analysis and Data Structures CS 6378, Advanced Operating Systems CS 6386, Telecommunications Software Design CS 6380, Distributed Computing CS 6390, Advanced Computer Networks CS 6392, Mobile Computing Systems
OTHER ASSIGNED DUTIES:	Faculty Senate, 2003-2004 PhD Committee, Computer Science, UTD, 2003-date Faculty Search Committee, 2001-2002, 2004-2005 Chair, MS-R committee, Computer Science, 2004-2005 Telecommunications Engineering Curriculum Development and Admissions Committee, 1998-2001 Computer Science Curriculum Committee, 1997-1998
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Balaji Raghavachari
DATE OF BIRTH:	
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, Pennsylvania State University, 1992 MS, Computer Science, Pennsylvania State University, 1992 BTech, mechanical Engineering, Indian Institute of Technology, Madras, 1984.
YEARS IN SERVICE AT UT DALLAS:	12 years
RELATED EXPERIENCE:	Post-Doctorate, John-Hopkins University, 1992-1993 Graduate Assistant, Penn State University, 1987-1992. Deputy Engineer, Bharat Electronics Ltd., India 1984-1987.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	R. Jothi and B. Raghavachari, "Degree-Bounded Minimum Spanning Trees," to appear in *Proceedings 16th Canadian Conference on Computational Geometry* (CCCG), 2004. R. Jothi and B. Raghavachari, "Approximation Algorithms for the Capacitated Minimum Spanning Tree Problem and its Variants in Network Design," *Proceedings 31st International Colloquium on Automata, Languages and Programming* (ICALP), 2004. R. Jothi and B. Raghavachari, "Improved Approximation Algorithms for the Single-Sink Buy-At-Bulk Network Design Problems," *Proceedings 9th Scandinavian Workshop on Algorithm Theory* (SWAT), 2004. V. Vokkarane, J. Wang, R. Jothi, X. Qi, B. Raghavachari, and J. Jue "Dynamic Dual-Homing Protection in WDM Mesh Networks," *Proceedings IEEE International Conference on Communications* (ICC), 2004. R. Jothi and B. Raghavachari, "Minimum Latency Tours and the k-Traveling Repairman Problem," *Proceedings Latin American Theoretical Informatics* (LATIN), pages 423-433, 2004. P. Gubbala and B. Raghavachari, "Finding k-Connected Subgraphs with Minimum Average Weight," *Proceedings Latin American Theoretical Informatics* (LATIN), pages 212-221, 2004. O. Daescu, R. Jothi, B. Raghavachari, and K. Sarac, "Optimal Placement of NAK Suppressing Agents for Reliable Multicast: A Partial Deployment Case," *Proceedings 19th ACM Symposium on Applied Computing* (SAC), pages 334-338, 2004. R. Jothi and B. Raghavachari, "Survivable Network Design: The Capacitated Minimum Spanning Network Problem," *7th*

	INFORMS Telecommunications Conference, 2004 (accepted for publication in *Information Processing Letters*) K. Deen, R. Jothi and B. Raghavachari, "Multi-Homing Protection in WDM Mesh Networks," *7th INFORMS Telecommunications Conference*, 2004. R. Jothi and B. Raghavachari, "Revisiting Esau-Williams' Algorithm: On the Design of Local Access Networks," *7th INFORMS Telecommunications Conference*, 2004. R. Jothi and B. Raghavachari, "Placement of Proxy Servers to Support Server-Based Reliable Multicast," *Proceedings 3rd IEEE International Conference on Networking* (ICN), 2004. R. Jothi and B. Raghavachari, "Dynamic Capacitated Minimum Spanning Trees," *Proceedings 3rd IEEE International Conference on Networking* (ICN), 2004
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	ACM
HONORS & AWARDS:	Outstanding Service Award, Erik Jonsson School, UTD, 2003. Outstanding Teacher in CS/UTD, 1999 August and Ruth Homeyer Fellowship, Penn. State, 1991. National Talent Search Scholarship (India), 1979-1984. Scholarship, National Merit Scholar (India), 1978.
COURSES TAUGHT 2000-2003:	CS 4347 Database Systems CS 6360 Database Design CS 6363 Computer Algorithms
OTHER ASSIGNED DUTIES:	University Scholarship Committee, 2004-2005. Assistant Chair, Department of Computer Science, UTD. 2003-2005 Member of PhD and MS-R Committees. 2003-2005. Computer Security Committee. 2003-2004. Chair of Computer Equipment Committee, 2001-2005.
SPECIFIC ROGRAMS INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Member of Editorial Board, Journal of Graph Algorithms and Applications, 1999-2004.

NAME:	David Russo
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	MS Computer Science, Southern Methodist University, 1996 BT Information Systems, University of North Florida, 1982
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Chief Technology Officer, Global Systems. Responsible for all technical decisions regarding design, development and support of a real-time marketing support system, 2001 – 2002 Senior Architect, Forelogic LLC. In this position I designed and led the implementation of a very large (> 150 k LOC) distributed system using J2EE infrastructure components, 2000 – 2001 Senior Engineer, Open Connect Systems. Worked as a lead-engineer for the development of 'screen-scraper' systems that allowed web applications to access legacy IBM programs, 1998 - 2000 Senior Software Engineer, Texas Instruments. Developed real-time missile guidance and signal processing software for a variety of weapon systems, 1983 - 1998
CONSULTING, PATENTS, ETC.:	Developed coursework design and implementation for Network Protocol and access employing the .NET Framework for Microsoft Inc., 2003 – current. Network design and implementation for Aramco Field Services, 2002 - 2003 Software Architecture consulting for Tetra-Pak system design, 2003
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	Reviewer and Consultant on "Distributed Computing, Principles and Applications, M. L. Liu, *Pearson Addison-Wesley*, ISBN 0-291-79644-9 (2003) "Concurrent Spiral Process", Joanne Jeska, David Russo, *Proceedings of the Raytheon Software Conference*, 1998.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Association for Computing Machinery
HONORS & AWARDS:	
COURSES TAUGHT 2000-	SE 4486 - SE Project

2003:	
OTHER ASSIGNED DUTIES:	UTD representative for the MSAA (Microsoft Academic Alliance) 2003 - present. Faculty Advisor for UTD Chapter of the ACM. Coach for a UTD ACM Programming Team
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Copyrights, Patents and Trademark course at SMU, 2002. .NET Professional Development Course 2003. ACM Software Engineering SIG Membership

NAME:	Martha A. Sanchez
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	MS, Computer Science (Multimedia and Graphics), George Washington University, 1997 BS, Computer Science, Universidad Autonoma Metropolitana, Mexico City, Mexico, 1989
YEARS IN SERVICE AT UT DALLAS:	4 years
RELATED EXPERIENCE:	UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO, UNAM 1998 – 1999, Mexico City/D.F., Mexico Web / CBT Manager, Computing Academic Services http://entrenate.dgsca.unam.mx •Formed and led a team developing and promoting Internet training courses, which required expertise in interfaces, user evaluations, UNIX administration, and multimedia tools. •Promoted web-based educational software and organized courses and workshops on this topic, aimed at teachers. •Performed research and wrote four papers on "Applied Learning Theories to WBT and CBT." •Wrote, organized, and developed story boards for web sites, multimedia and educational software for the Internet and CD-ROM. •Developed WBT software for the Internet in mathematical models, introductory computing, and software evaluation. •Established department goals and workflow for Computing Academic Services. •Interviewed and hired technical staff and consulted on distance education projects. SENSORY COMPUTING, Arlington, Virginia 1997
CONSULTING, PATENTS, ETC.:	CACTUN ESPACIO DIGITAL, Mexico City/D.F., Mexico 1999 – 2000 Independent Consultant, Multimedia / Web / UNIX http://www.cactun.com •Served as an advisor for Procter & Gamble's commercial CD-ROM interfaces. •Designed and implemented four commercial web sites utilizing the four main phases of development (www.cactun.com/tequilas, autos, cidem, and galeria). •Trained SGI and SUN Mexico in the UNIX operating system and various web tools; trained Morphos in Macromedia Director 6 and Flash 5.
STATE(S) IN WHICH	

REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	“Interactive multimedia sampler of learning theories.” Computación Visual, 1998. México, April, 1998. “Cursos por Internet.” Facultad de Psicología de la UNAM, Coloquio Comunidades de Aprendizaje: Un reto para la Universidad del siglo XXI. México, February, 1998. “Del Aula al Software Educativo.” 2° Foro de cómputo aplicado a la enseñanza.” FES Zaragoza. México, August, 1998. “Tecnologías de comunicación interactiva como facilitadores del proceso enseñanza aprendizaje en la Educación Superior.” Universidad de San Carlos de Guatemala. Guatemala, June, 1998. “Del Aula al Software Educativo.” Coordinación de Universidad Abierta y Educación a Distancia, Programa Universidad en Línea. México, June, 1998. “Evaluación de Software Educativo.” Haciendo Cursos en Línea, Coordinanción de Universidad Abierta y Educación a Distancia. México, October, 1998.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	•First Prize, Manuals / Course Materials Contest, Universidad Nacional Autonoma de Mexico (UNAM), for “Mathematica” (1991) and “UNIX System Administration” (1994) •Awarded, Grants totaling $105,000, Cray Research Scholarship Plan (to teach and promote the supercomputer and visualization topics for undergraduate students), 1992, 1993, 1994 •Scholarship Recipient, National University of Mexico, for the Master of Science in Computer Science degree program at George Washington University, 1995-1997
COURSES TAUGHT 2000-2004:	CS 2315 - Computer Science 2 (using Java) CS 3375/5375 - Principles of Unix CS 3336 Programming in Java CS3333 Data Structures CS5303 C/C++ CS5343 Data Structures CS5336 - Programming Projects in Java
OTHER ASSIGNED DUTIES:	Undergraduate Faculty Advisor CS/SE
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Kamil Sarac
DATE OF BIRTH:	August 11, 1971
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of California at Santa Barbara, 2002. MS, Computer Science, University of California at Santa Barbara, 1997. BS, Computer Engineering, Middle East Technical University, Turkey, 1994.
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Research Assistant, Computer Science, UC Santa Barbara. 1998-2002. Teaching Assistant, Computer Science, UC Santa Barbara. 1997-2000.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Kamil Sarac and Kevin C. Almeroth, “Tracetree: A Scalable Mechanism to Discover Multicast Tree Topologies in the Internet,” *IEEE/ACM Transactions on Networking, Vol 12, No. 5, pp.795-808*, October 2004. Kamil Sarac and Kevin C. Almeroth, “A Distributed Approach for Monitoring Multicast Service Availability,” *Journal of Network and Systems Management, Special Issue on Distributed Management, Vol. 12, No. 3, pp.327-348*, September 2004. Kamil Sarac and Kevin C. Almeroth, “Application Layer Reachability Monitoring for IP Multicast,” *accepted for publication in Elsevier Computer Networks Journal* (accepted in November 2004). Kamil Sarac and P. Namburi, “Multicast Session Announcements on top of SSM,” *IEEE International Conference on Communication*, Paris, France, June 2004. O. Daescu, R. Jothi, B. Raghavachari, and K. Sarac, “Optimal Placement of NAK Suppressing Agents for Reliable Multicast: A Partial Deployment Case”, *Proc. 19th ACM Symposium on Applied Computing*, Nicosia, Cyprus, March 2004. Kamil Sarac and Pavan Namburi, “End User Level Classification of Multicast Reachability Problems,” *2nd Workshop on End-to-End Monitoring Techniques and Services*, San Diego, CA, 2004.
SCIENTIFIC AND PROFESSIONAL	Association for Computing Machinery IEEE Computer Society

SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	QAD Fellowship Turkish Government fellowship, 1996-1998.
COURSES TAUGHT 2000-2004:	CS 4390: Computer Networks CS 4396: Computer Networks Lab CS 5390: Computer Networks CS 6390: Advanced Computer Networks CS 7301: Recent Advances in Computing: Advanced Network Services
OTHER ASSIGNED DUTIES:	Chair of publicity Committee in CS department.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Co-Chair, Special Track on Computer Networks, ACM Symposium on Applied Computing, 2004. Workshop Co-Chair, IEEE International Conference on Pervasive Services, 2005.

NAME:	Haim Schweitzer
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, Hebrew University, Israel, 1986 BS, Mathematics & Computer Science, Tel Aviv University, Israel, 1982
YEARS IN SERVICE AT UT DALLAS:	14 years
RELATED EXPERIENCE:	MTS, David Sarnoff Research Center, 1988-1990 Weizmann Post-Doctorate Fellowship, Computer Science, Cornell University, 1987-1988 Weizmann Post-Doctorate Fellowship, Computer Science, Columbia University, 1986-1987 Weizmann Post-Doctorate Fellowship, Computer Science, University of Texas at Austin, 1986
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	T. Yoshizawa and H. Schweitzer. "Interactive Browsing of Visual Content on the Internet". *Journal of Internet Technology*, page in press, 2005. T. Yoshizawa and H. Schweitzer. "Long-term learning of semantic grouping from relevance feedback". *Proc. of Sixth International Workshop on Multimedia Information*, 165-172, 2004 H. Schweitzer. "Computing Content-plots for video. *In A. Hayden, G. Sparr, M. Nielsen and P. Johansen, editors, Computer Vision ECCV 2002*, LNCS 2353, 491-501, Springer-Verlag, 2002. H. Schweitzer, J. W. Bell and F. Wu. "Very fast Template matching". *In A. Hayden, G. Sparr, M. Nielsen and P. Johansen, editors, Computer Vision ECCV 2002*, LNCS 2353, 358-372, Springer-Verlag, 2002. H. Schweitzer. "Template Matching Approach to content based image indexing by low dimensional Euclidean embedding". *Proc. of the International Conference on Computer Vision*, vol. 2, 566-571, IEEE Computer Society Press, July 2001. J. W. Bell and H. Schweitzer. "Determining face location in video conferencing applications". *Proc. of the International Conference on Imaging Science, Systems and Technology*, 522-525, CSREA Press, June 2001.
SCIENTIFIC AND	

PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	The Chaim Weizmann Postdoctoral Fellowship
COURSES TAUGHT 2000-2004:	Artificial Intelligence Machine Learning Computer Vision Introduction to Computer Vision
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Dr. Edwin (Hsing-Mean) Sha
DATE OF BIRTH:	
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, Princeton University, 1992 MA, Computer Science, Princeton University, 1991 BSE, Computer Science, National Taiwan University, 1986
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Assistant Professor, Dept. of Computer Science & Engr., University of Notre Dame, 1992-1995 Research Assistant and Teaching Assistant, Dept. of Computer Science, Princeton University, 1988-1992 System Programmer, Marine Corps, Taiwan, 1986-1988
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Z. Shao, Q. Zhuge and E. H.-M. Sha, ``Efficient Assignment and Scheduling for Heterogeneous DSP Systems," accepted in *IEEE Transaction on Parallel and Distributed Systems*. Y. Jiang, A. Al-Sheraidah, Y. Wang, E. H.-M. Sha and J. Chung, ``A Novel Multiplexer-Based Low-Power Full Adder," in *IEEE Transactions on Circuits and Systems II*, Vol. 51, No. 7, July 2004, pp. 345-348. T. W. O'Neil, and E. H.-M. Sha, ``Combining Extended Retiming and Unfolding for Rate-Optimal Graph Transformation," Accepted for Publication in *Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology*. Z. Shao, Q. Zhuge, M. Liu, C. Xue, E. H.-M. Sha and B. Xiao, ``Algorithms and Analysis of Scheduling for Loops with Minimum Switching," Accepted for Publication in *International Journal of Computational Science and Engineering (IJCSE)*, No. 2, 2004. Q. Zhuge, B. Xiao, and E. H.-M. Sha, ``Code Size Reduction Technique and Implementation for Software-Pipelined DSP Applications," in *ACM Transactions on Embedded Computing Systems (TECS)*, Vol. 2, No. 4, Nov. 2003, pp. 590-613.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	ISCA, ACM and The Institute of Electrical and Electronics Engineers (IEEE)
HONORS & AWARDS:	

COURSES TAUGHT 2001-2004:	Information Security, Parallel Architectures and Systems, Synthesis and Optimization of High-Performance Systems, Data Structures, VLSI Processor Arrays, Principles of Parallel Computing, Specialized Parallel Architectures, Operating Systems Principles, Automata
OTHER ASSIGNED DUTIES:	School Personnel Review Committee, ECS, University of Texas at Dallas, 2004 - 2006. Committee on Effective Teaching, University of Texas at Dallas, Representative member for ECS school, 2003 - 2005. Coordinator, Computer Systems Group, Department of Computer Science, University of Texas at Dallas, Jan. 2002- Present. Committee on Academic Affairs, Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, 2003 - Present. Committee on Effective Teaching, Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, 2002 - Present. University Internal Research Committee, University of Texas at Dallas, 2002 - Present. Committee on Educational Policy, University of Texas at Dallas, August 2001 - August 2003. Founding Co-director, *Hardware/Software Co-Design Lab for DSP and Communications,* University of Texas at Dallas, 2000. Founding Co-director, *Computer and Network Architecture Lab.* University of Texas at Dallas, 2000. Associate Chair of the Department of Computer Science, University of Texas at Dallas, May 2001 - Dec. 2001. Chair, Ph.D. Degree Program Committee of the Department of Computer Science, University of Texas at Dallas, Sept. 2000 - August 2001.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	G. Cort Steinhorst
DATE OF BIRTH:	October 1, 1946
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	**Masters of Science** in Computer Science, University of Southwestern Louisiana, Lafayette, LA (1973) **Bachelor of Science** in Liberal Arts (Mathematics with English minor), University of Southwestern Louisiana, Lafayette, LA. (1968)
YEARS IN SERVICE AT UT DALLAS:	5 Years
RELATED EXPERIENCE:	**Senior Lecturer** (Jan., 2000 -), Computer Science Program, School of Engineering and Computer Science, University of Texas at Dallas, Richardson, TX. **Computer Support Consultant** (Aug., 1999 – Jan., 2000), Grace Presbyterian Church, Plano, TX. **Manager** (July, 1982 – Aug., 1999), Atlantic Richfield Corporation (ARCO), Plano, TX. **Data Processing Analyst** (Sept., 1980 – July, 1982), Atlantic Richfield Corporation (ARCO), Dallas, TX. **Assistant Manager** (Jan., 1978 – Sept., 1980), Information Services, Texas Tech University, Lubbock, TX. **Computer Support Manager** (May, 1975 – Jan., 1978), Biomedical Engineering, Texas Tech University School of Medicine, Lubbock, TX. **Instructor** (Sept., 1973 – May, 1975), Computer Science, Texas Tech University, Lubbock, TX. **Instructor** (Sept., 1972 – Aug., 1973), Mathematics, Arkansas College
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	
COURSES TAUGHT 2000-	CS 1315, CS 2325, CS 3336. CS 2336: Computer Science II

2003:	CS 3333: Data Structures CS 3345: Data Structures and Introduction to Algorithmic Analysis CS 3385: Ethics, Law, Society, and Computing
OTHER ASSIGNED DUTIES:	Supervisor of CS 1115 Computer Science I Laboratory. Spring, 2003 had 3 lab sections. Fall, 2003 had 7 lab sections Recruited and supervised 7 Senior Mentors. The Senior Mentors either taught sections of the CS I Lab or assisted CS I instructors in their class room activities. Continued a tutoring laboratory for CS I students by utilizing assigned office hours of the Senior Mentors. The tutoring facility is open during the Fall and Spring semesters. Created a new set of Java oriented laboratory assignments for use in CS 1115
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Ivan H. Sudborough
DATE OF BIRTH:	
ACADEMIC RANK:	Founders Professor for Engineering and Computer Science
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD in Computer Science, Pennsylvania State University. 1971 MS in Mathematics, California State University at Hayward. 1967. BS in Mathematics, California State Polytechnic University at San Luis Obispo. 1966.
YEARS IN SERVICE AT UT DALLAS:	19 years
RELATED EXPERIENCE:	Professor, electrical Engineering and Computer Science, Northwestern University. 1971-85. Distinguished Visiting Professor of Math/Statistics, Miami University of Ohio, 1991 and 1998-1999. Distinguished Visiting Professor of Computer Science, University of Victoria (Canada) 1988. Fullbright Senior Research Professor, National Technical University of Greece, 1982-83. Visiting Professor of Mathematics/Computer Science, University of Paderborn (Germany), 1979-1980
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	I. H. Sudborough (with Z. Miller, M. Perkel, D. Pritikin), "The sequential sum problem and performance bounds on the greedy algorithm for the on-line Steiner problem", Networks 45(3):143-164, 2005. Ivan Hal Sudborough (with Linda Morales, M. Hossain Heydari and Mohamad J. Eltoweissy), "Combinatorial Optimization of Multicast Key Management", Journal of Networks and Systems Management, March 2004, pp. 32-50. Ivan Hal Sudborough (with Wolfgan Bein, Larry Larmore and Charles Shields), "Embedding a Complete Binary Tree into a Three-Dimensional Grid", Journal of Interconnection Networks, 2004 5(2), pp. 111-130. Ivan Hal Sudborough (with Wolfgan Bein, Larry Larmore and Sharom Latifif), "Block Sorting is Hard", International Journal of Foundations of Computer Science", 14(3)(2003), pp. 425-437. Ivan Hal Sudborough and Douglas Bass, "Pancake Problems with Restricted Prefix Reversals and some Corresponding Cayley Networks", Journal of Parallel and Distributed Processing (3) (2003), pp. 327-336. Ivan Hal Sudborough and Douglas Bass, "Hamilton

	Decompositions and (n/2)-Factorizations of Hypercubes", Journal of Graph Algorithms and Applications, vol. 7, no. 1, 2003, pp. 79-98. Ivan Hal Sudborough (with Y. B. Lin, Zevi Miller, Manley Perkel and Daniel Pritikin), "Expansions of Layouts of Complete Binary Trees into Grids", Discrete Applied Mathematics, 31, 2003, pp. 611-642.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	
COURSES TAUGHT 2000-2003:	CS 6363 Computer Algorithms
OTHER ASSIGNED DUTIES:	Computer Science Program Head, 1987-1995. Advisory Council, Erik Jonsson School of Engineering and Computer Science, 1987-1992 Chairman, Computer Science Curriculum Committee. 1996-98. Computer Science Graduate Admissions and Financial Aid Committee. Search Committee for Dean of Erik Jonsson School of Engineering and Computer Science (1986 and 1995) PhD Qualifying Examination Committee.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Program Committee Vice-Chair for Algorithms and Application for ISPAN 2004. Guest Editor for the International Journal of Foundations of Computer Science for the best papers of ISPAN 2004. General Chair, ISPAN 2002. General Chair, ISPAN, 2000. Editorial Board Member, Journal on Interconnection Networks (JOIN).

NAME:	Anthony D. Sullivan
DATE OF BIRTH:	December 1, 1938
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	B.S. Chemistry, Fordham University, 1960 M.S. Automatic Data Processing, George Washington University, 1970 Ph.D. Management Science, UTD, 1982
YEARS IN SERVICE AT UT DALLAS:	Part Time 14 Full Time 3
RELATED EXPERIENCE:	Lecturer, UTD School of Management, 1987-2002 31 years of experience as a corporate executive, manager and consultant. (including IBM, EDS, Texas Instruments, and Applied Data Research)
CONSULTING, PATENTS, ETC.:	Consulted with clients in utilities, health care, insurance, financial, retail, manufacturing, military, local and central government Certified as an IBM Information Technology Executive Consultant Inventor of record for a patent, (U.S. Patent Number 5,953,528), for “Knowledge Object Registration”, and has a patent pending for the ‘Innovation Assessment’. Extensive experience in the Department of Defense. He served 10 years on active duty with the U.S. Army, and 20 years in the reserve establishment. He retired with the rank of Colonel, having spent the last 9 years of his duty as the commander of a Strategic Intelligence Detachment, specializing in computer and network security. U.S. Patent Number 5,953,528 for “Knowledge Object Registration” U.S. Patent Number 6,820,071 for "Knowledge Management System and Method" Patent pending for the ‘Architecture for Knowledge Sharing’
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	
COURSES TAUGHT 2000-	BA 3351 Management Information Systems

2003:	CS 4V95 Requirements Engineering CS 6356 Software Reuse CS 6359 OOAD CS 6361 Requirements Engineering CS 6387 CASE CS 6V81 Individual Directed Studies MIS 6352 Web Design SE 3354 Software Engineering SE 4351 Requirements Engineering SE 4352 Software Architecture
OTHER ASSIGNED DUTIES:	ABET Course Sponsor
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	IBM Rational SEED IBM Scholars Program

NAME:	Laurie Thompson
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	MS, Computer Science, University of Texas at Dallas, 1998 BS, Building Construction, Texas A&M University, 1987
YEARS IN SERVICE AT UT DALLAS:	4 years
RELATED EXPERIENCE:	Lecturer , Computer Science, University of Texas at Dallas, 1999-2001 Network Support, Electronic Data Systems, 1990-1991 Engineer, Linbeck Construction Co, Ft. Worth, TX, 1987-1988
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	
COURSES TAUGHT 2000-2004:	CS 4340: Computer Architecture CS 1336: Programming Fundamentals
OTHER ASSIGNED DUTIES:	Undergraduate Faculty Advisor Member of Java Conversion Committee Collaborated on and modified ABET objectives for Computer Networks and Mathematics Maintain and modify ABET objectives for CS 2305.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Bhavani M Thuraisingham
DATE OF BIRTH:	
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D. in Theory of Computation and Computability Theory, University of Wales, United Kingdom, 1979. M.S. in Computer Science, University of Minnesota, 1984. M.Sc. in Mathematical Logic and Foundations of Computer Science, University of Bristol, United Kingdom, 1977. B.Sc. in Pure Mathematics, Applied Mathematics, and Physics University of Ceylon, 1975.
YEARS IN SERVICE AT UT DALLAS:	1 year
RELATED EXPERIENCE:	Co-founder, Data and Applications Security Program and Cyber Trust theme at NSF. Department Head, Data and Information Management, MITRE. Intergovernmental Personnel Act (IPA) at the National Science Foundation from the MITRE Corporation. Visiting Professor, New Mexico Institute of Technology. Adjunct Professor of Computer Science, University of Minnesota. Adjunct Professor of Computer Science, Boston University. Chief Scientist, Data Management, MITRE. Expert consultant in information security and data management to the Department of Defense, the Department of Treasury and the Intelligence Community for over 10 years. Instructor for AFCEA (Armed Forces Communication and Electronics Association) since 1998.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	*Privacy Sensitive Data Mining: Developments and Directions*, Journal of Database Management, 2005 (special issue in Database technologies for National Security) *Standards for Secure Semantic Web*, Computer Standards and Interface Journal (North Holland), March 2005 *Privacy Constraint Processing in a Privacy-Enhanced Database Management System*, Accepted for publication in Data and Knowledge Engineering Journal (North Holland), 2005. *A New Intrusion Detection System Using Support Vector Machines and Hierarchical Clustering*, Accepted in VLDB Journal, 2005 (co-authors: L. Khan, M. Awad)

Secure Third Party Publication of XML Documents, IEEE Transactions on Knowledge and Data Engineering, October 2004 (coauthor: E. Bertino et al)

Secure Sensor Information Management, IEEE Signal Processing, May 2004.

Managing and Mining Multimedia Databases, in AI Tools Journal (World Scientific), September 2004.

Security and Privacy for Sensor Databases, Sensor Letters, Inaugural Issue (American Scientific), Volume 2, #1, March 2004

Collaborative Commerce and Knowledge Management, Knowledge Management Journal (Wiley Interscience: coauthors: A. Gupta, E, Bertino et al), 2002

Secure Distributed Database Systems, Information Security Journal (Elsevier Science) special issue in database security, 2001

Real-time Priority Ceiling Algorithm, Real-time Systems Journal (Kluwer), 2001 (co-author: V. Wolfe, L. DiPippo et al)

Standards for Data Mining, Computer Standards and Interface Journal (North Holland) (coauthor: C. Clifton) 2001

Real-time CORBA, IEEE Transactions on Parallel and Distributed Systems (coauthor: V. Wolfe et al), October 2000.

Books

Web Data Management and Electronic Commerce, CRC Press, June 2000.

Managing and Mining Multimedia Databases for the Electronic Enterprise, CRC Press, June 2001

XML, Databases and the Semantic Web, CRC Press, March 2002.

Web Data Mining Technologies and Their Applications to Business Intelligence and Counter-terrorism, CRC Press, June 2003

Database and Applications Security: Integrating Information Security and Data Management, CRC Press, May 2005

SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:

IEEE

AAAS

HONORS & AWARDS:

Recipient of IEEE (Institute of Electrical and Electronics Engineers) Computer Society's 1997 Technical Achievement Award for contributions to Secure distributed database management. As Cited by IEEE, "this award is given to individuals who have made outstanding and innovative contributions in the field of computer and information science and engineering within the past 15 years."

Recipient of Career Communication Inc.'s National 2001 Woman of Color Technology Research Leadership Award

Featured by Silicon India's May 2002 issue as one of the top 7 technology innovators (only woman) in USA of South Asian origin (others are from Stanford, Berkeley, MIT, NASA, PARC and HP Labs). My innovation was for data and web security.

Recipient of IEEE's 2003 Fellow Award for Contributions to Secure Systems involving databases, distributed systems and the web. As stated by IEEE, "each year, following a rigorous evaluation procedure, the IEEE Fellow Committee recommends a select group of recipients for one of the Institute's most prestigious honors, election to IEEE Fellow".

Recipient of AAAS' (American Association for the Advancement of Science) 2003 Fellow Award for Contributions to Secure Web Information Systems.

Recipient of BCS' (British Computer Society) 2005 Fellow Award for Contributions to Data Security and Information Technology.

COURSES TAUGHT 2000-2005:	CS 6V81: Data and Applications Security
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Klaus Truemper
DATE OF BIRTH:	
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Operations Research, Case Western Reserve University, 1973 MS, Operations Research, Case Western Reserve University, 1972 MS, Industrial Engineering, University of Iowa, 1969 BS, Industrial Engineering, Rheinische School of Engineering, Germany, 1965
YEARS IN SERVICE AT UT DALLAS:	32 years
RELATED EXPERIENCE:	Visiting Professor, Dept. of Statistics, Tel-Aviv University, Israel, 1990 Visiting Professor, Applied Mathematics, University of Augsberg, Germany, 1990-1991 CNR-IASI, Rome Italy, 1998
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Klaus Truemper, *Design of Logic-based Intelligent Systems*, Wiley, to appear Spring 2005. Klaus Truemper, *Effective Logic Computation*, Wiley, New York, 1998. Klaus Truemper, *Matroid Decomposition*, Academic Press, Boston, 1992; revised edition Leibniz, Plano, Texas, 1998. G. Felici, A. Remshagen, and K. Truemper, "The Futile Questioning Problem," working paper, University of Texas at Dallas, May 2003. G. Felici, G. Rinaldi, A. Sforza, and K. Truemper, "Traffic Control: A Logic Programming Approach and a Real Application," *Ricerca Operativa* 30 (2001) A. Remshagen and K. Truemper, "Algorithms for Logic-Based Abduction," working paper, University of Texas at Dallas, 2002. H. Al-Mubaid and K. Truemper, "Learning to Find Context-Based Spelling Errors," working paper, University of Texas at Dallas, 2001. A. Remshagen and K. Truemper, "Learning in a Compiler for MINSAT Algorithms," *Theory and Practice of Logic Programming*, to appear.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A	

MEMBER:	
HONORS & AWARDS:	Faculty Development Award, UTD, 1997
COURSES TAUGHT 2000-2004:	Intelligent Systems Expert Systems Artificial Intelligence
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	R. N. Uma
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D. (Computer Science), Polytechnic University, Brooklyn, NY. 2000. M.E. (Computer Science), Indian Institute of Science, India. 1994. B.Sc. (Mathematics), University of Madras, India. 1990.
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Research Assistant, Polytechnic University, 1997, 1999-2000. Teaching Assistant, Polytechnic University, 1994-1997, 1998-1999.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	"Energy-Aware On-line Algorithms for Image Transmission over Wireless LAN", Chandramouli, S. Sri Ganesh Veera Kumar and R. N. Uma. *IEEE Interaction Conference on Communications* (ICC)}, June 2004. "On the Complexity and Hardness of the Steganography Embedding Problem", R. Chandramouli, Shalin Trivedi and R. N. Uma. *SPIE's 16th Annual Symposium Electronic Imaging Science and Technology*, January 2004. "Techniques for Scheduling with Rejection", Daniel W. Engels, David R. Karger, Stavros G. Kolliopoulos, Sudipta Sengupta, R. N. Uma and Joel Wein. *Journal of Algorithms* 49 (2003). (A special issue devoted to papers selected from ESA'98.) (Preliminary version appeared in *Proceedings of the 6th Annual European Symposium on Algorithms*, August 1998. Gianfranco Bilardi, Giuseppe F. Italiano, Andrea Pietracaprina and Geppino Pucci (eds.): Algorithms - ESA '98, LNCS 1461, Springer: Berlin, 1998.) "Dynamic Wavelength Assignment for Multicast in All-Optical WDM Networks to Maximize the Network Capacity", Jianping Wang, Biao Chen and R. N. Uma. *IEEE Journal on Selected Areas in Communications* (JSAC) 21(8) (October 2003). "Task Planning with Transportation Constraints: Approximation Bounds, Implementation and Experiments", Ovidiu Daescu, Derek Soeder and R. N. Uma. *Proceedings of the IEEE International Conference on Robotics and Automation* (ICRA), Sept. 2003. "An Experimental Study of LP-Based Approximation Algorithms for Scheduling Problems", Martin W.P. Savelsbergh, R. N. Uma

	and Joel Wein. *INFORMS Journal on Computing*, accepted August 2003. (Preliminary version appeared in *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms*.) "To Transmit or Not to Transmit: An Investigation Using Competitive Analysis", R. Chandramouli and R. N. Uma. *Proceedings of the IEEE Wireless Communications and Networking Conference*, March 2003.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	ACM, IEEE
HONORS & AWARDS:	Sun Microsystems Inc. Academic Equipment Grant. Clark Foundation Research Initiation Grant (Administered by Erik Jonsson School of Engineering and Computer Science at UT, Dallas.) Leadership Award: Advisor of the Year 2002-2003, University of Texas at Dallas. Pearl Brownstein Doctoral Research Award (Polytechnic University), 1999, for doctoral research showing great promise in the field of Computer Science. Deborah Rosenthal, M.D. Award (Polytechnic University), 1996, for outstanding performance on the Computer Science Ph.D qualifying examination. GATE (Graduate Aptitude Test in Engineering) Scholarship (a national level scholarship), 1992-1994. Indian Institute of Science Merit Scholarship, 1990-1992
COURSES TAUGHT 2001-2004:	CS 3345: Algorithm Analysis and Data Structures CS 4349: Advanced Data Structures and Algorithms CS 5333: Discrete Structures CS 7301/6V81: Recent Advances in Scheduling Theory
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Technical committee member for the Multimedia session at the IEEE International Conference on Communications (ICC) 2003. Technical program committee member for IEEE International Workshop on Online Algorithms for Mobile Wireless Computing and Networking (GLOBECOM 2004). Reviewer for Acta Informatica, European Symposium on Algorithms (ESA 2003), IEEE International Conference on Communications (ICC 2003, 2004), IEEE INFOCOM (2004), IEEE Transactions on Circuits and Systems for Video Technology, INFORMS Journal on Computing, International Conference on Supercomputing (ICS'97), International Journal of Computers and their Applications (a publication of ISCA), Journal of Algorithms, Journal of Scheduling, SIAM Journal on Computing (SICOMP) and Theoretical Computer Science A.

NAME:	Nancy Van Ness
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	MS, Mathematics, Stanford University BS, Applied Mathematics, Brown University , magna cum laude
YEARS IN SERVICE AT UT DALLAS:	8 years
RELATED EXPERIENCE:	Visiting Professor, Mathematics, Seattle University Visiting Professor , Mathematics and Director of Computer Science Concentration University of Dallas
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	
HONORS & AWARDS:	
COURSES TAUGHT 2000-2004:	CS 2305: Discrete Mathematics for Computing I CS 3305: Discrete Mathematics for Computing II CS 4384: Automata Theory(undergraduate) CS 5349: Automata Theory(graduate)
OTHER ASSIGNED DUTIES:	Graduate Advisor Advisor, Go Club
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Subbarayan Venkatesan
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of Pittsburgh, 1988 MS, Computer Science, University of Pittsburgh in 1985 MTech, Indian Institute of Technology, Madras, 1983 BTech, Indian Institute of Technology, Madras, 1981
YEARS IN SERVICE AT UT DALLAS:	15 years
RELATED EXPERIENCE:	
CONSULTING, PATENTS, ETC.:	``Method and system for restoring a distributed telecommunications network" U.S. Patent 5,999,286, Issued December 1999.
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	S. Venkatesan, K. Chandran, S. Raghunathan and R. Prakash, ``A Feedback Based Scheme For Improving TCP Performance In Ad-Hoc Wireless Networks',' *IEEE Communications Magazine, February 2001*. S. Venkatesan and S. Alagar, ``Techniques to Tackle State Explosion in Global Predicate," *IEEE Transactions on Software Engineering*, Volume 27, Number 8 August 2001 S. Venkatesan, M. Patel, and S. Chandrasekaran, "Efficient Minimum-Cost Bandwidth-Constrained Routing in Wireless Sensor Networks," *Proceedings of International Conference on Wireless Networks*, June 2004, to appear. S. Venkatesan, S. Krishnamurthy, M. Dawande and R. Chandrasekaran, "Highly Efficient Spare Capacity Planning for Generalized Link Restoration," *Proceedings of 12th International Conference on Computer Communications and Networks*, 2003, IEEE to appear. S. Venkatesan, M. Patel and R. Chandrasekaran, "A Comparative Study of Restoration Schemes and Spare Capacity Assignments in Mesh Networks," *Proceedings of 12th International Conference on Computer Communications and Networks*, 2003, IEEE to appear.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Secretary, IEEE Computer Society, Dallas Chapter, 1995
HONORS & AWARDS:	
COURSES TAUGHT 2000-	CS 5348/4348 Operating Systems

2004:	CS 6390 Advanced Computer Networks
OTHER ASSIGNED DUTIES:	Chair, UTD Intellectual Property Committee, 2002-date. Member, Faculty Senate, 2003-date Group Coordinator, Networking Group, 2002-date Chair, Colloquium Committee, 1997-1998 Dean Search Committee, 2002-2003 Governing Committee, Telecom Engineering Program, 2002-date Department Reorganization Committee, 2001 Faculty Search Committee, 2003-date; Chair, 1996-1997. Building Planning Committee, 2003
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Yuke Wang
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of Saskatchewan, Canada, 1996. MSc, Mathematics, University of Saskatchewan, Canada, 1992. BSc, Mathematics, University of Science and Technology, Hefei, China, 1989.
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Assistant Professor, Florida Atlantic University, 1999-2000. Assistant Professor, Concordia University, Canada, 1996-1999. Post-Doctoral Fellow, University of Montreal, Canada, 1995-1996. Programmer Analyst, SED System Inc., Saskatoon, Canada, 1993.
CONSULTING, PATENTS, ETC.:	**Patents:**Yuke Wang, Anand Krishnamurthy, Lie Qian, Philippe dauchy and Alberto Conte, “Load Adaptive Router in QoS Architectuure”, **Alcatel France**, March 2003. Lie Qian, Anand Krishnamurthy, Yuke Wang, Yiyan tang, Phillipe Dauchy and Alberto Conte, “S-BIND Traffic Model and Gamma H_BIND Admission Control Algorithm on On-Line Traffic”, **Alcatel France**, March 2003. Yuke Wang, Yun Zhang, Yiyan Tang, Anand Krishnamurthy, Lie Qian and Gerard Damm, “Disjoint Graph Based Classification Algorithm for range_Specified Rules”, **Alcatel Canada**, August 2003. Lie Qian, Yiyan Tang, Yuke Wang, B. Bou-Diab and W. Olensinski, “Dynamic and Static Tunneling Schemes for Scalable Mulitcast in MPLS Network” with **Alcatel Canada**, November 2004.
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	X. Song, G. Yang, M. Perkowski and Yuke Wang, “Algebraic Characterization of Reversible Logic Gates”, *Theory of Computing Systems*, December 2004, pp. 1-9. Y. Jiang, Yuke Wang, Y. Savaria and X. Song, “Computation of Signal Output Probability for Boolean Functions Represented by OBDD”, *International Journal of Computers and Mathematics with Applications*, vol. 47, 2004, pp. 1865-1874. Y. Jian, Abdul Karim Al-Sheraidah, Yuke Wag, Edwin Sha and Jin-gyun Chung, “A novel low power multiplexer-based full adder”, *IEEE Transactions on Circuits and Systems-II*, July 2004, vol. 51, Issue:7, pp.345-348

	Wei Wang, M.N. S. Swamy, O. Ahmad, Yuke Wang, "Comprehensive VLSI study of residue-binary arithmetic conversion", *IEEE Transactions on Circuits and Systems*, February 2003, vol. 50, issue. 2, pp. 235-243.
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE
HONORS & AWARDS:	
COURSES TAUGHT 2000-2003:	Computer Architecture Computer Networks Custom VLSI design Design and Analysis of Algorithms Design and Implementation of DSP algorithms DSP architecture File Organize Method Introduction to Object -Oriented Sys.
OTHER ASSIGNED DUTIES:	January 2000 – April 2000, Chair selection committee, Department of Computer Science, Florida Atlantic University, to select a new chair for the department. September 2000 – now, Graduate Admission and TA committee, Department of Computer Science, University of Texas at Dallas, to admit M. Sc. and Ph. D. students and to assign Teaching Assistantship to qualified students. April 2001 – now, Computer Engineering Committee, Department of Computer Science, University of Texas at Dallas, to jointly manage the computer engineering program in the college. September, 2001 – now, Faculty recruiting committee, Department of Computer Science, University of Texas at Dallas, to recruit new faculty members.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Editor, *IEEE Transactions on VLSI*, published by IEEE Circuits and Systems Society. Editor, *IEEE Transactions on Circuits and Systems*, published by IEEE Circuits and Systems Society. Editor, *Journal of Circuits, Signals, and Systems*, published by Birkhäuser Boston. Editor, *Journal of Applied Signal Processing*, published by Hindawi Publishing Corporation. *International Journal of Parallel and Distributed Systems & Networks*, published by ACTA Press.

NAME:	Wei Wei
DATE OF BIRTH:	
ACADEMIC RANK:	Senior Lecturer
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Juris Doctor Candidate, SMU Dedman School of Law Ph.D. in Computer Science Telecommunications, University of Missouri-KC, 1993 M. Eng. in Computer Science, Chinese Academy of Sciences, Beijing, China, 1986 B.S. in Computer Science, Zhongnan University, Hunan, China, 1983
YEARS IN SERVICE AT UT DALLAS:	2 years
RELATED EXPERIENCE:	Manager, Systems Engineering and QA Testing, Iris Labs, 2000-2001 Sr. Team Lead Systems Engineer, Alcatel USA, 1998-2000 Principle Engineer, IEX Corporation, Richardson, Texas, 97-98 Senior Software Engineer, TCSI Corporation, Dallas,1995-1997 Consultant, NEC AM, Advanced Switching Laboratory, Irving, TX 1994 – 1995 Lecturer, Computer Science Telecom, Univ. of Missouri at Kansas City 1993 – 1994 Project Leader, OSE Laboratory/AlliedSignal 1990 – 1993 Research Associate, Chinese Academy of Sciences, Beijing, CHINA 1984 - 1988
CONSULTING, PATENTS, ETC.:	Telecom Consulting, 2003-2004 White Rock Networks. On SONET (UPSR, BLSR, VT Grooming) and Ethernet (VLAN) systems and software design and testing. Telecom Consulting, 2001-2002 Azuola Systems. On WiFi based ordering systems and software design and testing. Telecom Consulting, 1998-1999 ATT. On RPP (Rapid Provisioning Platform) systems modeling and design. Telecom Consulting, 1995-1997 GTE. On CABS (Carrier Access and Billing Systems) systems modeling and design. Telecom Consulting, 1994-1995 NEC/Sprint. On ATM network management systems design and implementation. Efficient Flow Control Mechanism for Event Traffic (EU Patent. Pending in US. Co-inventor) Network Element Intelligence for Alarm/Event Processing (First-inventor) Web Server Based Nodal Management (Co-inventor) Super-User Link Between Intra-Management Domains (First-inventor) User Interface to Hide Transaction (First-inventor) Inter-Area and Inter-Autonomous System Management (Co-

	inventor) Service Affecting Events/Alarms Treatment Architecture (First-inventor) Efficient Access Control Mechanism for Alarm/Event Delivery (First-inventor) Using Spread Scheduling to Avoid Network Management Traffic Bottleneck (First-inventor) Alarm Summarization for Efficient Alarm Synchronization (First-inventor) Application Navigation with Context Passing (First-inventor) Efficient Performance Data Collection Unifying Operator Information Across A Network (First-inventor)
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS I NTHE LAST FIVE YEARS:	Directed Intelligence: Policy Management and LDAP (OSScon'01)
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	State Bar of Texas / LSD
HONORS & AWARDS:	Recipient of University Missouri Special Chancellor's Award for 1990 - 1991. Winner of INTEROP'92 Networking Achievement Award in Education Category
COURSES TAUGHT	CS 1336 Fundamentals of Programming/C++ CS2305 Discrete Structures I CS 3305 Discrete Structures II CS 3345 Algorithm Analysis and Data Structures CS 4349 Advanced Data Structures and Algorithms CS 6V81 Personal Communication Systems CS 6V81 Bluetooth & Ad Hoc Networks CS 6385 Algorithms/Telecom Network Design CS 6386 Telecom Software Design CS 6360 Database Designs CS 6390 Advanced Computer Networks CS 420 Internetworking: Bridges and routers (UMKC) CS 490 Client/Server Programming (UMKC) CS 520 Advanced Computer Networks (UMKC)
OTHER ASSIGNED DUTIES:	Graduate Faculty Advisor
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING &	

PROFESSIONAL COMPETENCE:	

NAME:	Weichen Eric Wong
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science. Purdue University MS, Computer Science, Purdue University BS, Computer Science, Eastern Michigan University
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Sr. Research Scientist and Project Manager, Telcordia Technologies (formerly Bellcore).
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	W. E. Wong, T. Sugeta, J. J. Li, and J. Maldonado, "Coverage Testing Software Architectural Design in SDL," *Journal of Computer Networks*, Volume 42, Issue 3, pp. 359-374, June 2003 W. E. Wong, T. Sugeta, Y. Qi, and J. C. Maldonado, "Smart debugging software architectural design in SDL," *Journal of Systems and Software* (accepted for publication) W. E. Wong and J. J. Li, "Redesigning Legacy Systems into the Object-Oriented Paradigm," *International Journal of Software Engineering and Knowledge Engineering*, Volume 14, Number 3, pp. 255-276, June 2004 W. E. Wong and S. Gokhale, "Static and Dynamic Distance Metrics for Feature-Based Code Analysis," *Journal of Systems and Software*, Volume 74, Number 3, pp. 283-295, 2004 A. Vincenzi, J. Maldonado, W. E. Wong, and M. Delamaro, "Coverage Testing of Java Programs and Components," *Journal of Science of Computer Programming* (accepted for publication) S. S. Gokhale, W. E. Wong, J. R. Horgan, and K. S. Trivedi, "An Analytical Approach to Architecture-Based Software Performance and Reliability Prediction," *Journal of Performance Evaluation* (accepted for publication) F. Liu, W. Guo, W. Chou, and W. E. Wong, "An Approach of Integrating SIP in Converged Multimodal/Multimedia Communication Services," *Journal of Telecommunication Systems* (accepted for publication) J. Cangussu, K. Cooper, and W. E. Wong, "An Empirical

	Evaluation of a Run-Time Dynamic Adaptable Framework," *The Journal Studia Informatica Universalis* (accepted for publication)
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE
HONORS & AWARDS:	Quality Assurance Special Achievement Recognition, NASA/Johnson Space Center, 1997
COURSES TAUGHT 2000-2004:	CS 6354 Advanced Software Engineering CS 5333 Discrete Structure SE 4367 Software Testing, Verification, Validation and Quality Assurance CS 2305 Discrete Math for Computing I CS 3305 Discrete Math for Computing II
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	**Journal Guest Editor** A special issue of the Journal of Software Practice and Experience (SPE) on *Developing Trustworthy Software Systems* (in preparation) A special issue of the International Journal of Software Engineering and Knowledge Engineering (IJSEKE) on *Aspect-Oriented Software Design* (in preparation) A special issue of the Journal of Software Practice and Experience (SPE) on *Enhancing Network Applications*, Volume 33, Issue 14, November 2003. **General Chair** ICCCN 2003 - The 12th IEEE International Conference on Computer Communications and Networks, Dallas, Texas, October 20-22, 2003 **PC Chair** ISSRE 2005 - The 16th IEEE International Symposium on Software Reliability Engineering, Chicago, Illinois, November 8-11, 2005 SEKE 2005 - The 17th International Conference on Software Engineering and Knowledge Engineering, Taipei, Taiwan, July 14-16, 2005 COMPSAC 2004 - The 28th IEEE International Computer Software and Applications Conference, Hong Kong, September 28-30, 2004 ICCCN 2002 - The 11th IEEE International Conference on Computer Communications and Networks, Miami, Florida, October 14-16, 2002 **Operation Committee Chair** COMPSAC 2003 - The 27th IEEE International Computer Software and Applications Conference, Dallas, Texas, November

3-6, 2003

QSIC 2003 - The 3rd International Conference on Quality Software, Dallas, Texas, November 6-7, 2003

Workshop Chair

COMPSAC 2005 - The 29th IEEE International Computer Software and Applications Conference, Edinburgh, Scotland, July 25-28, 2005

NAME:	Weili (Lily) Wu
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	Ph.D. 2002, Computer Science and Engineering, University of Minnesota. M.S. 1997, Computer Science and Engineering, University of Minnesota.
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Research & Teaching Assistant, Computer Science, University of Minnesota, 1996-2002 Research & Teaching Assistant, Economics, University of Wisconsin, 1994-1995 China Coal Research Academic, 1989-1993
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	D.Z. Du, P.M. Pargalos, & W. Wu, *Mathematical Theory of Optimization*, (Springer Verlag) Z, Zhang, W. Wu, & Y. Huang, "Mining Dynamic Interdimension Association Rules for Local-scale Weather Prediction," *SIAM Workshop on Mining Scientific and Engineering Datasets*, 2004. H. Goa, J. Li, Y. Li, & W. Wu, "Computing Cube on Compressed Datasets in Data Warehouses," *VLDB*, 2004. H. Gao, J. Li, Y. Li, & W. Wu, "New algorithm for computing cube on very large compressed datasets," *IEEE Transactions on Knowledge and Data Engineering*, 2003 W. Wu, C. Li, X. Huang, & Y. Li, "On error-fault tolerant DNA screening," *Journal of Combinatorial Optimization*, 2004
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE Computer Society American Association for the Advancement of Science Association for Computing Machinery
HONORS & AWARDS:	UCGIS Summer Assembly Student Travel Award, 2000 Excellent Student Scholarship, Liaoning Technical University, 1986
COURSES TAUGHT 2001-2004:	Database Design Implementation of Database Management Systems Operating Systems
OTHER ASSIGNED DUTIES:	

SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:

NAME:	I-Ling Yen
DATE OF BIRTH:	
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of Houston, 1992. MS in Computer Science, University of Houston, 1985. BS in Physics, National Tsing-Hua Univesrity, 1979.
YEARS IN SERVICE AT UT DALLAS:	8 years
RELATED EXPERIENCE:	Assistant professor, Dept. of Computer Science, Michigan State University. 1992-1997. Sesimic Data Processing Programmer, Western Geophysical, Houston. 1990-1991. Software Quality Assurance Engineer, Valid Logic Systems Inc., San Jose. 1986-1988. Systems Programmer, COINS, University of Massachusetts, Amherst. 1985-86.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	*Journal Papers* Peng Li, I-Ling Yen, and Zhonghang Xia, “Preference update for E-commerce applications: Model, language, and processing,” accepted by *Electronic Commerce Research Journal*. Zhonghang Xia, Wei Hao, I-Ling Yen, Peng Li, “A distributed admission control model for QoS assurance in large-scale media delivery systems,” accepted by *IEEE Transaction on Parallel and Distributed Systems*. Manish Gupta, Manghui Tu, Latifur Khan, Farokh Bastani, and I-Ling Yen, “A study of the model and algorithms for handling location dependent continuous queries,” accepted by *International Journal on Knowledge and Information Systems*. Peng Li, I-Ling Yen, and Zhonghang Xia, “Optimizing concurrent M^3-transactions: A fuzzy constraint satisfaction approach,” accepted by *Journal on Systemics, Cybernetics and Informatics*. Ing-Ray Chen, Ngoc Anh Phan, I-Ling Yen, “Update propagation algorithms for supporting disconnected write in mobile wireless systems with data broadcasting capability,” accepted by *Journal of Wireless Personal Communications*. Peng Li, I-Ling Yen, and Zhonghang Xia, “M3-Update: a new update model for E-Commerce and web-based applications,” accepted by *International Journal of Computers and Their*

Applications.

Ing-Ray Chen, Sheng-Tun Li, and I-Ling Yen, "Adaptive QoS control based on benefit optimization for video servers providing differentiated services", *Multimedia Tools and Applications*, Vol. 25, No. 2, Feb 2005,pp. 167-185.

Feng Luo, Latifur Khan , F.B. Bastani, I-Ling Yen, and J. Zhou, "A dynamical growing self-organizing tree (DGSOT) for hierarchical clustering gene expression profiles," *Bioinformatics Journal*, Oxford University Press, Nov 2004, pp. 2605-2617.

Sung Kim, Farokh B. Bastani, I-Ling Yen, and Ing-Ray Chen, "Systematic reliability analysis of a class of application-specific embedded software frameworks," *IEEE Transactions on Software Engineering*, Vol. 30, No. 4, April 2004, pp. 218-230.

Dongfeng Wang, Farokh B. Bastani, and I-Ling Yen, "A systematic design method for high quality process-control systems development," *International Journal of Software Engineering and Knowledge Engineering*, Vol. 14, No. 1, February 2004, pp. 43-60.

D.C. Wang, I.R. Chen, C.P. Chu, and I.L. Yen, "Replicated object management with periodic maintenance in mobile wireless systems," *Journal of Wireless Personal Communications*, Vol. 28, Jan. 2004, pp. 17-33.

SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE
HONORS & AWARDS:	*A Distributed Component Repository for Rapid Synthesis of Adaptive Real-Time Systems*, National Science Foundation, 9/2001 8/2004. *Advanced Radar and Electro-optical Sensor Systems*, Army Space and Missile Defense Command (SMDC), 2002-2003. *Support for Adaptive Multi-Criteria Transaction Processing in E-Commerce Applications*, Texas Advanced Technology Program, Jan 2000 - Dec 2001. *Assessing Y2K Compliance for Mission-Critical Systems*, Army Research Laboratory, Sep 1, 1999 - August 31, 2000. *Research Experience for Undergraduates*, National Science Foundation, Sep 1, 1998 - August 31, 1999. *Establishing a Computer-Aided Education Environment using the Web Lecture System*, TxTEC, July 1, 1998 - August 31, 1999. *Establishing a Computer-Aided Education Environment using the Web Lecture System*, Nortel, July 1, 1998 - August 31, 2000. *Processor Specialization in Fault-Tolerant Distributed Systems*, National Science Foundation, Sep. 1, 1996 - Dec 31, 1999. *Systematic Integration of Fault Tolerance in High Performance*

	Parallel Programs, National Science Foundation, Sep. 1995 - June 1997. A Run-Time Support System for Scalable Object-Oriented Parallel Programming, GE Foundation, June 1994 - August 1995.
COURSES TAUGHT 2000-2004:	CS 4348 Operating Systems CS 6378 Advanced Operating Systems CS 6378 Advanced Operating Systems (Telecampus version) Information Assurance Embedded Software Multimedia Systems Web Technology
OTHER ASSIGNED DUTIES:	Member, Search Committee, Spring and Fall 2004. PhD Committee, Spring 2004. Member, Core Committee for the Support of Women and Minorities.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Panelist and Reviewer for NSF Program Co-Chair, 2005 International Symposium on Autonomous Decentralized Systems (ISADS’ 05) Program Co-Chair, 2005 International Workshop on Software and Compilers for Embedded Software (SCOPE’05)

NAME:	Kang Zhang
DATE OF BIRTH:	April 12, 1959
ACADEMIC RANK:	Associate Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of Brighton, UK, 1990 Certificate in English Language Training, Guangzhou Foreign Language Institute, China, 1985 BEng, Computer Engineering, University of Electronic Science and Technology, Chengdu, China
YEARS IN SERVICE AT UT DALLAS:	5 years
RELATED EXPERIENCE:	Lecturer, Senior Lecturer, Computing, Macquarie University, Sydney, Australia, 1991-1999 SERC Postdoctoral Research Fellow, Electrical and Electronic Engineering, UK Science and Engineering Council, University of Brighton, UK, 1990-1991 Research Assistant, Electrical and Electronic Engineering, University of Brighton, UK, 1986-1990 Software Engineer, CAD Section East China Research Institute of Computer Technology, Shanghai, China. 1982-1985
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	K. Zhang (Ed.), *Software Visualization - From Theory to Practice*, Kluwer Academic Publishers, Boston, April 2003, ISBN 1-4020-7448-4, 568 pages. K. Zhang, D-Q. Zhang, and J. Cao, “Design, Construction, and Application of a Generic Visual Language Generation Environment", *IEEE Transactions on Software Engineering*, Vol.27, No.4, April 2001, 289-307. N. Stankovic, D. Kranzlmueller, and K. Zhang, “The PCG: An Empirical Study”, *Journal of Visual Languages and Computing*, Academic Press, Vol.12, No.2, April 2001, 203-216. K. Zhang and G. Wirtz, “Issues in Visual Parallel and Distributed Programming”, *Journal of Visual Languages and Computing*, Academic Press, Vol.12, No.2, April 2001, 217-220. D-Q. Zhang and K. Zhang, and J. Cao, “A Context-Sensitive Graph Grammar Formalism for the Specification of Visual Languages”, *The Computer Journal*, Vol.44, No.3, Oxford University Press, 2001, 186-200. N. Stankovic and K. Zhang, “A Distributed Parallel Programming Framework”, *IEEE Transactions on Software Engineering*, Vol.28, No.5, May 2002, 478-493.

	J. Cao, A.T.S. Chan, Y. Sun, and K. Zhang, “Dynamic Configuration Management in Graph-Oriented Distributed Programming Environment”, *Science of Computer Programming*, Vol.48, No.1, July 2003, Elsevier Science Inc., 43-65. K. Zhang, J. Kong, M.K. Qiu, and G.L. Song, “Multimedia Layout Adaptation Through Grammatical Specifications”, *ACM/Springer Multimedia Systems*, Vol.10, No.3, 2004 (in press).
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	British Computer Society Association for Computing Machinery Association for Logic Programming Australian Computer Society IEEE
HONORS & AWARDS:	Outstanding Service Award, UTD, 2003
COURSES TAUGHT 2000-2004:	CS 2315 C/C++ CS 2325 Assembler CS 4361 Computer Graphics CS 5330 Computer Science II (Computer Organization) CS 6359 Object Oriented Analysis and Design CS 6366 Computer Graphics
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	

NAME:	Youtao Zhang
DATE OF BIRTH:	
ACADEMIC RANK:	Assistant Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science Department, University of Arizona, 2002. MEng, Computer Science Department, Nanjing University, 1996. BS, Computer Science Department, Nanjing University, 1993.
YEARS IN SERVICE AT UT DALLAS:	3 years
RELATED EXPERIENCE:	Research Assistant, University of Arizona, Tucson, 1999-2002. Summer Intern, Hewlett-Packard Company, May 1998-August 1998. Software Engineer, Pacific City Inc.(China), 1996-1997.
CONSULTING, PATENTS, ETC.:	
STATE(S) IN WHICH REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	Yongjing Lin, Youtao Zhang, Quanzhong Li, and Jun Yang, "Supporting Efficient Query Processing on Compressed XML Files," *ACM The 20th Annual Symposium on Applied Computing*, Santa Fe, New Mexico, March, 2005. Youtao Zhang, Lan Gao, Jun Yang, Xiangyu Zhang, and Rajiv Gupta, "SENSS: Security Enhancement to Symmetric Shared Memory Multiprocessors," *IEEE 11th International Symposium on High Performance Computer Architecture*, San Francisco, California, February 2005 Xiangyu Zhang, Rajiv Gupta, and Youtao Zhang, "Cost and Precision Tradeoffs of Dynamic Slicing Algorithms," *ACM Transactions on Programming Languages and Systems*, to appear. Youtao Zhang, and Jun Yang, "Reducing I-cache Energy of Multimedia Applications through Low-Cost Tag Comparison Elimination," *Journal of Embedded Computing*, to appear. Youtao Zhang, Jun Yang, Yongjing Lin, and Lan Gao, "Architectural Support for Protecting User Privacy on Trusted Processors," *The Workshop on Architectural Support for Security and Anti-Virus*, **In conjunction with the 11th ASPLOS*, Boston, MA, October 2004.*** Bengu Li, Youtao Zhang and Rajiv Gupta, "Speculative Subword Register Allocation in Embedded Processors," **The 17th International Workshop on Languages and Compilers for Parallel Computing**, *West Lafayette*, Indiana, September 2004. Youtao Zhang and Rajiv Gupta, "Enabling Partial Cache Line Prefetching Through Data Compression," **High-Performance**

Computing: Paradigm and Infrastructure, *John Wiley & Sons, Inc*. (to appear).
Guodong Li, Youtao Zhang, Yongjin Lin, Yaochun Huang, "Scalable Duplication Strategy with Bounded Availability of Processors," *IEEE The Tenth International Conference on Parallel and Distributed Systems, Newport Beach, California*, July 2004.
Xiangyu Zhang, Rajiv Gupta, and Youtao Zhang, "Efficient Forward Computation of Dynamic Slices Using Reduced Ordered Binary Decision Diagrams," ***IEEE/ACM International Conference on Software Engineering***, *Edinburgh, UK*, May 2004.

SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	IEEE, ACM, ACM SIGSOFT
HONORS & AWARDS:	NSF Career Award, January 2005. Distinguished paper Award, IEEE/ACM International Conference on Software Engineering, May 2003. Most Original Paper Award, International Conference on Parallel Processing, October 2003. The Elite Graduate Scholarship, Nanjing University China), June 1995. The Lupoe Graduate Scholarship, Nanjing University China), June 1994 The Guanghu Scholarship, Nanjing University (China), October 1992.
COURSES TAUGHT 2000-2004:	CS 5348 Operating System CS 6353 Compiler Construction
OTHER ASSIGNED DUTIES:	
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	PC member, web chair, ACM/IEEE International Symposium on Code Generation and Optimization (CGO), 2005. PC Member, IASTED International Conference on Parallel and Distributed Computing And Networks (PDCN), 2005. PC Member, ACM SIGPLAN Conference on Languages, Compilers and Tools for Embedded Systems (LCTES), June 2004. NSF ITR panelist, 2003.

NAME:	Si Q Zheng
DATE OF BIRTH:	
ACADEMIC RANK:	Professor
DEGREES WITH FIELDS, INSTITUTIONS AND DATES:	PhD, Computer Science, University of California, Santa Barbara, 1987. MS in Mathematical Sciences, University of Texas at Dallas, 1982. BS in Electrical Engineering, Jilin University, China. 1973.
YEARS IN SERVICE AT UT DALLAS:	7 years
RELATED EXPERIENCE:	Adjunct Associate Professor, Electrical and Computer Engineering, Louisiana State University. 1995-1998. Tenured Associate Professor, Computer Science, Louisiana State University. 1993-1998. Assistant Professor, Computer Science, Louisiana State University. 1987-1993. Graduate Assistant, University of California at Santa Barbara. 1984-1987. Graduate Assistant, University of Texas at Dallas. 1980-1984.
CONSULTING, PATENTS, ETC.:	Consultant, Alcatel, 2000 -2001. Consultant, Telchip, 2000-2001. Consultant, Bonanza.com Inc., 1999-2000. S.Q. Zheng, Y. Xiong, and Marc Vandenhoute, Hardware Implementation of Channel Scheduling Algorithms for Optical Routers with FDL Buffers. Pending US patent. S.Q. Zheng, Y. Xiong, and Steve Y. Sakalian, Unified Associate Memory of Data Channel Schedulers in an Optical Router, US Ser. No. 60/257,884, 12/22/2000. Y. Xiong and S.Q. Zheng, Channel Scheduling in Optical Routers, US Ser. No. 60/257,487, 12/22/2000. S.Q. Zheng and Y. Xiong, Optical Burst Scheduling Using Partitioned Channel Groups, US Ser. No. 60/257,382, 12/22/2000. S.Q. Zheng and Y. Xiong, Ingress Edge Router Architecture and Related Channel Scheduling Algorithms for OBS Networks. Pending US patent. S.Q. Zheng, J. Blanton, P. Golla, D. Verchere, and D. Zriny, A Parallel Round-Robin Arbiter for Switch Control. Pending US patent. Y. Yang, S.Q. Zheng, and D. Verchere, Group Switching for DWDM Optical Networks. Pending US patent. S.Q. Zheng, M. Yang, and F. Masetti-Placci, Programmable Parallel k-Selectors as Schedulers of Multiserver Systems. Pending US patent.
STATE(S) IN WHICH	

REGISTERED:	
PRINCIPAL PUBLICATIONS IN THE LAST FIVE YEARS:	S.Q. Zheng, C. Li and M. Yang, "Scalable Schedulers for High-Performance Switches," *Proceedings of 2004 IEEE Workshop on High Performance Switching and Routing* (HPSR), 2004. S.Q. Zheng, M. Yang, B. Bhagyavati and Stan Kurkovsky, "Programmable Weighted Arbiters for Constructing Switch Schedulers," *Proceedings of 2004 IEEE Workshop on High Performance Switching and Routing* (HPSR), 2004. S.Q. Zheng and E. Lu, "Parallel Routing and Wavelength Assignment for Optical Multistage Interconnection Networks," *Proceedings of the 33th International Conference on Parallel Processing* (ICPP-2004), 2004. S.Q. Zheng and M. He, "An Optimal Generalized Columnsort Algorithm on a 2D ARPBS," to appear in *Proceedings of the 16-th IASTED International Conference on Parallel and Distributed Computing and Systems*, 2004. S.Q. Zheng and Y. Yang, "Group Switching for DWDM Optical Networks," to appear in *Proceedings of the 13th International Conference on Computer Communications and Networks* (ICCCN), 2004. S.Q. Zhang, M. Yang, E. Lu and J. Wang, "Hierarchical Scheduling for DiffServ Classes" to appear in *Proceedings of IEEE Globecom 2004*. S.Q.Zheng and M. He, "An Optimal Mulitiway Mergesort Algorithm on a 2D ARPBS," to appear in *Proceedings of Computing and the 17-th ISCA International Conference on Parallel and Distributed Computing Systems*, 2004. S.Q. Zheng, E. Lu, M. Yang and B. Yang, "A Class of Self-Routing Strictly Nonblocking Photonic Switching Networks," to appear in *Proceedings of IEEE Globecom 2004*. S.Q, Zheng, C. Li, P. Zhai and B. Prabhakaran, "Segmentation and Recognition of Multi-attribute Motion Sequences," to appear in *Proceedings of ACM Multimedia 2004*. S.Q. Zheng, C. Li, G. Pradhan, and B. Prabhakaran, "Indexing of Variable Length Multi-attribute Motion Data," to appear in *Proceedings of the 2nd ACM International Workshop on Multimedia Databases* (MMDB 2004).
SCIENTIFIC AND PROFESSIONAL SOCIETIES OF WHICH A MEMBER:	Senior Member of IEEE. Member of IEEE Computer Society.
HONORS & AWARDS:	
COURSES TAUGHT 2000-2004:	Digital Logic Design (undergraduate) Introduction to Computer Science (undergraduate) Computer Architectures (undergraduate)

Programming Languages (undergraduate)
Discrete Mathematics for Computing (undergraduate)
Advanced Data Structures and Algorithms (undergraduate)
Discrete Structures (graduate)
Operating Systems (graduate)
Advanced Computer Architectures (graduate)
Advanced Operating Systems (graduate)
Introduction to Computational Geometry (graduate)
Algorithms for VLSI CAD (graduate)
Algorithmic Aspects of Telecommunication Networks (graduate)
Special Topics in Computer Science: Computational Geometry (graduate)

OTHER ASSIGNED DUTIES:	Member, Academic Senate, 2003-present. Associate Head, Computer Science, 2002-2004. Associate Head, Electrical Engineering, 2001-2002. Member, Dean of Engineering School Search Committee, 2002. Chairman, Committee on Qualifications, 2000-2001.
SPECIFIC PROGRAMS IN WHICH INVOLVED TO IMPROVE TEACHING & PROFESSIONAL COMPETENCE:	Chairman, *The 17th IASTED International Conference on Parallel and Distributed Computing and Systems*, 2005. Chairman, *The 14th ISCA International Conference on Parallel and Distributed Computing Systems*, 2001. Program Committee Chairman, *The 11th IASTED International Conference on Parallel and Distributed Computing and Systems*, 1999. Program Committee Vice Chairman, *The Second International Conference on Parallel and Distributed Computing and Networks*, 1998. Program Committee Chairman, *The 8th International Conference on Computing and Information*, 1996.

APPENDIX XI

COURSE DESCRIPTIONS

• ***Computer Science Course Description***

Department Course #	MATH 2417	**Course Title**	Calculus I
Total Credits	3	**Course Coordinator**	C.D. Cantrell
		URL, if any	

Current Catalog Description:

Functions, limits, continuity, differentiation; integration of function of one variable; logarithmic, exponential, and inverse trigonometric functions; techniques of integration, and applications. Three lecture hours and two discussion hours a week. Prerequisite: A SAT II Mathematics Level IC Test score of 560, a Level II Test score of 530, or a grade of at least C in MATH 2312 or an equivalent course. (4-0) S

Textbook(s), References and/or Other Required Material:

CALCULUS by Larson, Hostetler and Edwards, 7th Edition, Student Solution Manual Available in Bookstore

References

WEB: The utility WebCT, accessible from a UTD computer lab or from your own Web connection, will be an essential communication tool for this course. The
URL is http://webct.utdallas.edu; this can also be accessed from the UTD home page.

Course Goals/Objectives:

After completing this course the student will be able to differentiate and integrate simple functions and will be able to apply calculus to scientific and engineering problems**.**

Prerequisite:

MATH 2312 or an equivalent course

Major Topics Covered in the Course:

- Preparation for Calculus (Self-review for students)
- Limits and their Properties
- Differentiation
- Applications of Differentiation
- Integration
- Logarithmic, Exponential, and Other Transcendental Functions
- Applications of Integration
- Integration Techniques, L'Hôpital's Rule, and Improper Integrals

Projects:

None.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations	X	
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course aims at introducing the student to the basic concepts of calculus.

Problem Analysis and Design:

Addressed by the assignments.

Prepared/Reviewed:

Prepared: C.D. Cantrell
Reviewed: Gopal Gupta & Simeon Ntafos

• *Computer Science Course Description*

Department Course #	MATH 2418	**Course Title**	Linear Algebra
Total Credits	3	**Course Coordinator**	C.D. Cantrell
		URL, if any	

Current Catalog Description:

Systems of linear equations, determinants, vectors and vector spaces, linear transformations, eigen values and eigenvectors, quadratic forms. Three lecture hours and two discussion hours per week. Credit given for only one of MATH 2333 or 2418.

Textbook(s), References and/or Other Required Material:

Elementary Linear Algebra, Howard Anton, (eighth edition).

References:

None

Course Goals/Objectives:

After completing this course the student will have a basic understanding of fundamental techniques of linear algebra. The concepts of vector spaces, matrices, determinants, inverses, eigen values and eigenvectors will be developed.

Prerequisite:

MATH 2419 or an equivalent course

Major Topics Covered in the Course:

- Systems of Linear Equations and Matrices
- Determinants
- Vectors in 2-Space and 3-Space
- Euclidean Vector Spaces
- General Vector Spaces
- Inner Product Spaces
- Eigen values and Eigenvectors
- Linear Transformations

Projects:

None

CSAB Category Content:

	CORE	ADVANCED

Theoretical Foundations	X	
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course aims at introducing the student to the basic concepts of linear algebra.

Problem Analysis and Design:

Addressed by the assignments.

Prepared/Reviewed:

Prepared: C.D. Cantrell
Reviewed: Gopal Gupta & Simeon Ntafos

• *Computer Science Course Description*

Department Course #	MATH 2419	**Course Title**	Calculus II
Total Credits	3	**Course Coordinator**	C.D. Cantrell
		URL, if any	

Current Catalog Description:

Continuation of MATH 2417. Improper integrals, sequences, infinite series, power series, parametric equations and polar coordinates, vectors, vector-valued functions, functions of several variables, partial derivatives and applications, multiple integration. Three lecture hours and two discussion hours a week. Prerequisite: A score of at least 4 on the Advanced Placement Calculus BC exam or MATH 2417. (4-0) S

Textbook(s), References and/or Other Required Material:

CALCULUS by Larson, Hostetler and Edwards, 7th Edition, Student Solution Manual Available in Bookstore

References:

WEB: The utility Web CT, accessible from a UTD computer lab or from your own web connection, will be an essential communication tool for this course. The URL is http://webct.utdallas.edu; this can also be accessed from the UTD home page.

Course Goals/Objectives:

To provide an understanding of series convergence and approximation, plane and space curves, and of vectors, partial differentiation and multiple integration, and their applications in the sciences.

Prerequisite:

MATH 2417 or an equivalent course

Major Topics Covered in the Course:

- Integration Techniques, L'Hôspital's Rule, and Improper Integrals
- Infinite Series
- Conics, Parametric Equations, and Polar Coordinates
- Vectors and the Geometry of Space
- Vector-Valued Functions
- Functions of Several Variables
- Multiple Integration

Projects:
None

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:
None

Social and Ethical Issues:
Issues related to enforcing scholastic honesty.

Theoretical Content:
The course aims at introducing the student to advanced concepts in calculus.

Problem Analysis and Design:
Addressed by the assignments.

Prepared/Reviewed:
Prepared: C.D. Cantrell
Reviewed: Gopal Gupta & Simeon Ntafos

- ***Computer Science Course Description***

Department Course #	PHYS 2325/PHYS 2125	Course Title	***Mechanics and Heat/Physics LaboratoryI***
Total Credits	3/ 1	**Course Coordinator**	***C.D. Cantrell***
		URL, if any	

Current Catalog Description:

PHYS 2325 Mechanics and Heat Calculus based. Basic physics including a study of space and time, kinematics, forces, energy and momentum, conservation laws, rotational motion, torques, and harmonic oscillation. Two lectures and one recitation session per week. Prerequisite: MATH 2417. Co-requisite: PHYS 2125. (3-0) Y

PHYS 2125 Physics Laboratory I Laboratory course to accompany PHYS 2325. Personal computer-based data presentation and curve fitting. Basic measurement concepts such as experimental uncertainty, mean, standard deviation, standard error, and error propagation will be covered. Co-requisite: PHYS 2325. (0-3) Y

Textbook(s), References and/or Other Required Material:

University Physics (11th edition) Volume 1, Young and Freedman.
Online HW: Mastering Physics for Young/Freedman, 11th edition

References:

N/A

Course Goals/Objectives:

This is a list of what I expect you to know and be able to do by the end of this class; Addition, scalar multiplication, and vector multiplication of vectors, Understand the components of linear motion (displacement, velocity, acceleration), Understand the different forces and work force problems, Understand Newton's laws of motion, Understand the different types of energy, Use the conservation of energy to work problems, Understand impulse, momentum and collisions, Understand center of mass and rigid bodies motion, Know rotational variables and the relationship between linear and rotational variables, Be able to solve problems using rotational and linear variables, Understand and work with equilibrium situations including the different types of equilibrium, Understand simple harmonic motion and waves including their properties, Understand fluids in motion and at rest, Understand heat and heat transfer mechanisms, Understand the three laws of thermodynamics, Know the types of engines and refrigerators

Prerequisite:

Math 2417-Calculus I

Major Topics Covered in the Course:

- Introduction, Units, Vectors
- Velocity, Acceleration, 1-D Motion,2-D and 3-D Motion, Constant Acceleration
- Newton's Laws and applications
- Work, Potential Energy
- Gravity
- Momentum, Impulse, Collisions in 1-D,Collisions in 2-D
- Angular Motion
- Moment of Inertia
- Torque, Rolling, Angular Momentum
- Static Equilibrium
- Simple Harmonic Motion
- Waves, Pressure
- Buoyancy, Ideal Gas
- First Law of Thermodynamics, PV Diagrams
- Second Law of Thermodynamics, Engines, Refrigerators

Projects:

The course includes experiments designed to explore several areas in Mechanics and Heat. The experiments in PHYS 2125 give you the opportunity to examine several phenomena in great detail. In Physics, there is interplay between the theory that you see in a class or read about and experimental work. One is not more important than the other but one informs the other: theoretical predictions are a natural focus of experiment and experimental results help to develop theory.

It is the responsibility of the student in this lab course to familiarize herself/himself with concepts required in any experiment. The manual is a rough guide to concepts required and not a thorough explanation of those ideas. Books that fully describe those concepts are available.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations	X	
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course aims at introducing the student to the basic concepts mechanics and heat.

Problem Analysis and Design:

Addressed by the assignments and laboratory experiments.

Prepared/Reviewed:

Prepared: C.D. Cantrell
Reviewed: Gopal Gupta & Simeon Ntafos

- ***Computer Science Course Description***

Department Course #	PHYS 2326/PHYS 2126	**Course Title**	Electromagnetism and Waves/ Physics Laboratory II
Total Credits	3/1	**Course Coordinator**	C.D. Cantrell
		URL, if any	

Current Catalog Description:

PHYS 2326 Electromagnetism and Waves (*3 semester hours*) Continuation of PHYS 2325. Topics include electrostatics and electromagnetics, electric field and potential, electric currents, magnetic fields, laws of Coulomb, Ampere, and Faraday, Maxwell's theory of propagation. Two lectures and one recitation session per week. Prerequisites: PHYS 2325 and MATH 2419. Co-requisite: PHYS 2126. (3-0) Y

PHYS 2126 Physics Laboratory II (*1 semester hour*) Laboratory course to accompany PHYS 2326. Builds on concepts of Physics Lab I. Will emphasize the use of an oscilloscope and measurements using simple circuits constructed in class. Co-requisite: PHYS 2326. (0-3) Y

Textbook(s), References and/or Other Required Material:

The basic material is covered in many textbooks and students may use any of them. This particularly refers to: R.A. Serway and R.J. Beichner, **"Physics for Scientists and**

Engineers", Volume 2; D. Halliday, R. Resnick, J. Walker, **"Fundamentals of Physics"**, Volume 2.

References:

R.P. Feynman, R.B. Leighton, & M. Sands, **"The Feynman Lectures on Physics"**, Vol. 2 – Read this for the Physics of it! ;A. Shadowitz, **"The Electromagnetic Field".**

Course Goals/Objectives:

This is an introductory course on electricity, magnetism and electromagnetic waves. The goal is for students to develop an understanding and gain a practical knowledge of basic notions of electric charges, currents, electromagnetic fields and forces. Our focus is not on "training" and "dry" learning of the lecture materials but on conceptual understanding (broad concepts like "physical fields vs action-at-a-distance", "superposition principle", etc.) and developing skills to apply basic principles to actual problem solving. Lectures and problem solving sessions will include examples of how to approach problems; students are expected to spend as much as possible of their own time on problems, quizzes, etc. Some part of the lectures will be devoted to topics beyond the textbook content and intended to make students aware of more advanced stuff, to put things in a more general picture and to be, in a sense, inspirational. Lectures and other materials will be made available online.

Prerequisites:

PHYS 2325 and MATH 2419. Co-requisite: PHYS 2126. (3-0) Y

Major Topics Covered in the Course:

- Introduction to Electric Charges and Fields
- Properties of Electric Charges
- Coulomb's Law and Superposition;
- Conductors and Insulators;
- Magnetic fields and Lorenz force;
- Mathematics of Vector Fields
- Electrostatics in Vacuum and Dielectrics
- Electric Current and DC Circuits
- Magnetostatics in Vacuum and Matter
- Motion of Charges in Electric and Magnetic Fields
- Time-Dependent Fields and Currents
- Maxwell’s Equations and Electromagnetic Waves

Projects:

The course includes experiments designed to explore several areas in Electromagnetism and Waves. The experiments in PHYS 2126 give you the opportunity to examine several phenomena in great detail. In Physics, there is interplay between the theory that you see in a class or read about and experimental work. One is not more important than the other but one informs the other: theoretical predictions are a natural focus of experiment and experimental results help to develop theory.

It is the responsibility of the student in this lab course to familiarize herself/himself with concepts required in any experiment. The manual is a rough guide to concepts required and not a thorough explanation of those ideas. Books that fully describe those concepts are available.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course aims at introducing the student to the concepts of electromagnetism and waves.

Problem Analysis and Design:

Addressed by the assignments and laboratory experiments.

Prepared/Reviewed:

Prepared: C.D. Cantrell
Reviewed: Gopal Gupta & Simeon Ntafos

• ***Computer Science Course Description***

•

Department Course #	ECS 3390	**Course Title**	Professional and Technical Communication
Total Credits	3	**Course Coordinator**	C. D. Cantrell
		URL, if any	

•

Current Catalog Description:

• Course utilizes an integrated approach to writing and speaking for the technical profession. The writing component focuses on writing professional quality technical documents such as proposals, memos, abstracts, reports and letters. The oral communication part of the course focuses on planning, developing, and delivering dynamic, informative and persuasive presentations. Gives students a successful communication experience working in a functional team environment using a total on-line/real time learning environment.

Textbook(s), References and/or Other Required Material:

Specified by Instructor

Course Goals/Objectives:

After completing this course the student will be able too communicate effectively with customers, technical peers and management orally and in writing.

Prerequisite:

RHET 1302

Major Topics Covered in the Course:

Written and oral technical communications

Projects:

None.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations	X	
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

The course prepares the student to communicate effectively with customers, technical peers and management orally and in writing

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Specified by the instructor.

Problem Analysis and Design:

None

Prepared/Reviewed:

Prepared: C.D. Cantrell
Reviewed: Gopal Gupta & Simeon Ntafos

• *Computer Science Course Description*

Department Course #	ISSS 3360	**Course Title**	Politics and Values in Business and Technology
Total Credits	3	**Course Coordinator**	C.D. Cantrell
		URL, if any	

•

Current Catalog Description:

• A social and behavioral science survey of current business practices and the normative value systems by which they operate and are regulated. Topics will include the influences on business practices by culture, especially race, ethnicity, gender, religion, and by developing technology and the Information Society. (3-0) S

Textbook(s), References and/or Other Required Material:

John R. Boatright. *Ethics and the Conduct of Business* (Prentice Hall, 2000)

Tronto, Joan. Care as a Political Concept" from *Revisioning the Political: Feminist Reconstructions of Traditional Concepts in Western Political Thought* (ed.) Nancy Hirschmann and Christine DiStefano (Westview, 1996

Hirschman, Albert O. "Rival Views of Market Society" in *Rival Views of Market Society and Other Recent Essays* (Viking, 1986).

England, Paula. "Policy Debates." *Comparable Worth: Theories and Evidence* (Aldine de Gruyter, 1992) pp. 277-308.

Course Goals/Objectives:

• Develop an (i) understanding of broader contemporary business issues; (ii) understanding of professional and ethical responsibilities; (iii) understanding of how business practices change with cultural and societal evolution.

Prerequisite:
None

Major Topics Covered in the Course:

An Introduction to the Politics and Ethics of Business
Utilitarianism
Rights Based Ethics
Virtue Ethics and Ethics of Care
Politics, Values and the Concept of the Marketplace
Rival Views of Market Society

The Corporation
Sexual and Racial Discrimination
Pay Equity
Intellectual Property and Corporate Trade Secrets
Computer Information and Privacy
Health and Safety in Business and Technology
Global Capitalism and Neo-Colonialism

Projects:

Assigned paper readings.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations	X	
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to business and professional ethics.

Theoretical Content:

Specified by the instructor.

Problem Analysis and Design:

None

Prepared/Reviewed:

Prepared: C.D. Cantrell
Reviewed: Gopal Gupta & Simeon Ntafos

• ***Computer Science Course Description***

Department Course #	CS 1336 & CS 1136 Lab	**Course Title**	Programming Fundamentals
Total Credits	4 (not acceptable for degree credit in ECS)	**Course Coordinator**	Cort Steinhorst / Laurie Thompson
		URL, if any	Responsibility of Instructor e.g., *www.utdallas.edu/~csteinh*

Current Catalog Description:

CS 1336: *Introduction to Computers*. Primitive data types, variable declarations, variable scope, and primitive operations. Control statements. Methods/functions, including recursion. Arrays, including implementation of a stack data structure and strings using primitive data arrays. Output formatting. Debugging techniques. Designed for students with no prior computer programming experience.

CS 1136 Lab: *Laboratory for CS 1336*. This course teaches basic computer literacy/programming skills: disk operating system (DOS) commands (to format disks and to create, manipulate, and remove directories and files), the authoring of ASCII text files, compiler usage in converting source programs into executable form, printer commands.

Textbook(s), References and/or Other Required Material:

- *Starting Out with C++*, 4th Edition (Standard Version), 2005 Update by Tony Gaddis and Barret Krupnow.
- *Starting Out with C++*, 4th Edition (Lab Manual) by Dean DeFino and Michael Bardzell

References:

None

Course Goals/Objectives:

To develop the student's abilities to:

- Develop algorithmic solutions for use on computers
- Express algorithmic solutions in a high level computer language
- Use programming tools such as an editor, compiler, and linker
- Utilize fundamental programming structures - linear processing
- Utilize fundamental programming structures - conditional processing
- Utilize fundamental programming structures - loop processing
- Process data in arrays
- Develop programs in a functional/method form
- Perform searches and sorts
- Utilize reference variables
- Manipulate character and string data

Prerequisites:

None

Major Topics Covered in the Course:

Expressions & Interactivity, File Input & Output, Decisions, Looping, Functions, Arrays, Searching & Sorting, Pointers, Character Strings and Structures.

Projects:

Programming project(s) are assigned for each topic.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		
Algorithms		
Data Structures	X	
Programming Concepts	X	
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

None

Theoretical Content:

Basic Programming

Problem Analysis:

Basic Programming

Solution Design:

Basic Programming

Prepared/Reviewed:

Prepared: Cort Steinhorst
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• Computer Science Course Description

Department Course #	CS 1337	**Course Title**	Computer Science I
Total Credits	3	**Course Coordinator**	Cort Steinhorst
		URL, if any	Responsibility of Instructor e.g., *www.utdallas.edu/~csteinh*

Current Catalog Description:
Introduction to object-oriented software analysis, design, and development. Classes and objects. Object composition and polymorphism. Sorting, searching, recursion. Strings and stacks using core classes. Inheritance and interfaces. Graphic User Interfaces. Includes a comprehensive programming project.

Textbook(s), References and/or Other Required Material:
Introduction to JAVA Programming, 5th Edition by Y. Daniel Liang, Prentice Hall

References:
None

Course Goals/Objectives:
After successful completion of this course, the student should have an:
- Ability to develop object oriented software solutions for use on computers
- Ability to express algorithmic solutions in a high level computer language
- Ability to utilize the String classes
- Ability to utilize express multi-class relationships among objects
- Ability to implement graphical user interfaces
- Ability to develop graphical programs utilizing standard layout managers
- Ability to develop event driven programs
- Ability to process data with abstract data types
- Ability to perform searches and sorts
- Ability to develop programs utilizing recursive methodology
- Ability to utilize reference variables

Prerequisites:
CS 1336 or equivalent programming experience.

Major Topics Covered in the Course:
Methods, Recursion, Objects and classes, Strings, Class Inheritance & Interfaces, Object-oriented software development, GUI Programming, Sorting & Searching, and Files.

Projects:
Programming project(s) are assigned for each topic.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		
Algorithms	X	
Data Structures	X	
Programming Concepts	X	
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

None

Theoretical Content:

Basic Programming

Problem Analysis:

Basic Programming

Solution Design:

Basic Programming

Prepared/Reviewed:

Prepared: Cort Steinhorst
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS 2110	**Course Title**	Introduction to Digital Systems Laboratory
Total Credits	1	**Course Coordinator**	Herman Harrison
		URL, if any	Responsibility of each Faculty member. e.g., www.utdallas.edu/~herman.harrison

Current Catalog Description:

The purpose of the laboratory is to give students an intuitive understanding of digital circuits and systems. Laboratory exercises include construction of simple digital logic circuits using prototyping kits and board level assembly.

Textbook(s), References and/or Other Required Material:

- *Schaum's Outline of Theory and Problems of Digital Principles*, Roger L. Tokheim, Third Edition, McGraw-Hill.

References:

None

Course Goals/Objectives:

- Ability to recognize standard digital circuits and understand their usage
- Ability to design a combinational logic circuit based on a truth table requirements
- Ability to design a simple combinational logic circuit based on a boolean expression
- Ability to build and test a combinational logic circuit on a prototype board
- Ability to understand sequential logic elements, including flip-flops, latches, storage registers, shift registers, and counters
- Ability to build a sequential logic circuit from a requirements statement

Prerequisites/Corequesite:

CS 2310 – Introduction to Digital Systems.

Major Topics Covered in the Course:

This course is an accompanying course to CS 2310 and is designed to give hands-on experience to students on the concepts learnt in CS 2310. The topics covered are:

- Standard digital circuits
- Design of Combinational Logic Circuits
- Sequential logic elements including flip-flops, latches
- Building Sequential Logic circuits

Projects:

There will be no semester-long projects. Students are expected to conduct a separate experiment in each lab session.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations	X	
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture	X	

Oral and Written Communication:

Students are expected to submit reports detailing the experiments conducted in each lab session.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

None.

Problem Analysis:

The lab experiments require the student to solve problems bases on the requirement statement.

Solution Design:

The students ate expected to design and build combinational circuits and sequential circuits.

Prepared/Reviewed:

Prepared: Herman Harrison
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• Computer Science Course Description

Department Course #	CS 2305	**Course Title**	Discrete Mathematics for Computing I
Total Credits	3	**Course Coordinator**	Timothy Farage
		URL, if any	Responsibility of each Faculty member. e.g., http://www.utdallas.edu/~tfarage

Current Catalog Description:

Principles of counting. Boolean operations. Logic and methods of proof. Sets, relations, functions, strings, and languages.

Textbook(s), References and/or Other Required Material:

Discrete Mathematics and its Applications by Rosen, 5th Edition

References:

None

Course Goals/Objectives:

The ABET objectives for this course are:

- Ability to understand mathematical facts in order to read, comprehend and construct mathematical arguments
- Ability to use and apply basic definitions and properties of sets and logic
- Ability to specify precise meaning of mathematical statements, using quantifiers and predicates as needed
- Ability to recognize and construct valid proofs
- Ability to understand and use various types of functions
- Ability to understand and construct a proof by induction
- Ability to use modular arithmetic as it relates to computer science problems such as hashing and encryption
- Ability to recognize and use Boolean algebra applied sets and logic
- Ability to understand what an algorithm is and to use algorithms
- Ability to use basic counting techniques such as permutations, combinations
- Ability to write recursive definitions and function

Prerequisites:

MATH 1326 or MATH 2417 or consent of the instructor.

Major Topics Covered in the Course:

The topics to be covered are:

- **Logic, Sets and Functions**: connectives, truth tables, laws of logic, quantifiers, predicates, function.

- **Algorithms, Integers and Matrices**: algorithms, complexity of algorithms, integers, number theory.
- **Mathematical Reasoning and Recursion:** proofs, math induction, recursive algorithms.
- **Counting**: permutations, combinations, combinations with repetition, the binomial theorem.
- **Discrete Probability:** probability theory, expected value and variance.

Projects:

No Projects

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations	X	
Algorithms		
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

While there is no formal grading of oral or written communication, students are encouraged to ask questions in class and turn in homework in a neat and readable fashion.

Social and Ethical Issues:

The only ethical issue discussed in this course is that of the integrity of the degree be kept by limiting incidences of cheating on exams, quizzes and homework.

Theoretical Content:

This course covers a lot of an introduction of theory. It has in it the theory of recurrence relations and their solutions. Basic underlying abstract concepts of graphs and trees used in more advanced courses for modeling problems are also a large part of the course. The discussion of big O and other complexity issues are first introduced. More advanced techniques of counting are also developed which will be useful in the statistics required in the degree program. Students are urged to think in more abstract ways to apply solutions in different contexts and not just adapt small changes of known algorithms.

Problem Analysis:

Much of this course is devoted to problem analysis. Students are encouraged to apply underlying concepts in various situations. While no projects are assigned the logical approach to small problems first are encouraged as means of finding a more general solution is encouraged (particularly in the advanced counting techniques). Breaking down of problems into more manageable pieces is discussed. Students are encouraged to state the problem in their own words and understand thoroughly what the question is. This is done in the context of discrete mathematical problems.

Solution Design:

This course does not deal with aspects of design.

Prepared/Reviewed:

Prepared: Timothy Farage
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	CS 2310	**Course Title**	***“Introduction to Digital Systems”***
Total Credits	3	**Course Coordinator**	***Herman Harrison***
		URL, if any	Responsibility of each faculty member. e.g., www.utdallas.edu/~herman.harrison

Current Catalog Description:

Introduction to hardware structures and assembly-language concepts that form the basis of the design of modern computer systems. Internal data representation and arithmetic operations in a computer. Basic logic circuits. MIPS assembly language. Overview of PC architecture.

Textbook(s), References and/or Other Required Material:

1. *Schaum's Outline of Theory and Problems of Digital Principles*, 3rd Edition by Roger L. Tokheim, McGraw-Hill, ISBN 0-07-065050-0
2. *Introduction to RISC Assembly Language Programming* by John Waldron, Addison-Wesley, 1998

References:

Computer Organization and Design - The Hardware/Software Interface 2nd Edition by Patterson and Hennessey, Morgan-Kaufmann, 1998, ISBN 1-55860-428-6

Course Goals/Objectives:

The student should be able to implement basic digital logic and independently develop, test, and document assembly language programs demonstrating proficiency with each of the above features.

Specific objectives are:

- Utilize fundamental hardware concepts in modern computer systems
- Utilize fundamental of Boolean algebra
- Design basic logic circuits, combinational and sequential
- Process binary information and use 2's complement arithmetic
- Develop algorithmic solutions for use on computers
- Express algorithmic solutions in assembly language
- Utilize fundamental programming structures - sequential processing
- Utilize fundamental programming structures – conditional processing
- Utilize fundamental programming structures - loop processing
- Process data in arrays
- Develop programs in a modular form
- Utilize bit oriented logical instructions
- Demonstrate comprehension of floating point arithmetic
- Demonstrate comprehension of interrupt processing

- Demonstrate comprehension of single and multi-cycle CPU designs
- Demonstrate comprehension of pipeline architectures/hazards

Prerequisites:

CS 1337 – Computer Science I

Major Topics Covered in the Course:

1. Digital logic
2. Number representations and 2’s complement arithmetic
3. Assembly language programming and structures
4. Computer architectural concepts including RISC and pipelined CPU designs

Projects:

There where 6 programming assignments which cover (1) Assembly language and simulation, (2) Arithmetic, (3) Comparisons and branching, (4) Loops and iterations, (5) Subprograms, (6) Arrays and sorting.

There where 5 digital lab experiments which covered (1) Logic gates and/or/not, (2) Boolean expressions, (3) Digital adders, (4) Bistable circuits and flipflops (5) Shift registers and counters

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations	X	
Algorithms	X	
Data Structures		
Programming Concepts	X	
Computer Architecture	X	

Oral and Written Communication:

The 6 homework exercises provide the students the opportunity to improve their written communication skills. This is demonstrated through the documentation required with the assignment. The 5 laboratory exercises because of their size, 2,5,10pgs, provide additional experience in the organization and presentation of data. The students must demonstrate the ability to describe, interpret, and support/refute the results of the experiments in which they were a contributor.

Social and Ethical Issues:

None

Theoretical Content:

1. Digital logic including basic gates, truth tables, state machines

2. Numerical fundamentals of number representation, digital arithmetic including integer multiplication/division, and floating point arithmetic.
3. Computer architecture including control, datapath, memory, and I/O interfaces
4. CPU design including RISC, pipelines, and hazards

Problem Analysis and Design:

The homework problems are designed to provide the student experience in the area of processor level arithmetic, flow control, procedures, data structures (arrays/pointers), and device input/output.

Prepared/Reviewed:

Prepared: Herman Harrison
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS 2336	**Course Title**	Computer Science II
Total Credits	3	**Course Coordinator**	Lawrence King
		URL, if any	Responsibility of each Faculty member. e.g., www.utdallas.edu/~herman.harrison

Current Catalog Description:
Exceptions and number formatting. File input/output using Stream classes. Implementation of primitive data structures, including linked lists (all types), stacks, queues, and binary trees. Advanced data manipulation using core classes. Introduction to multithreading, multimedia, and networking. Includes a comprehensive programming project.

Textbook(s), References and/or Other Required Material:
Required: *Introduction to Java Programming*, 5th Edition by Y. Daniel Liang, Prentice Hall
Optional: *The Java Tutorial*, 3rd Edition by Campione, Walrath, & Huml, Addison Wesley

References:
None

Course Goals/Objectives:
- Ability to implement a comprehensive OO application
- Ability to create and use primitive data structures
- Ability to use core Java data structures – stack, queue, tree
- Ability to use core Java data structures – lists
- Ability to use core Java data structures – maps
- Ability to implement a GUI for user interaction
- Ability to create and use exception handlers
- Ability to create and use graphical error messages
- Ability to use file input/output – text files
- Ability to use file input/output – object files

Prerequisites:
CS 1337 – Computer Science I

Major Topics Covered in the Course:
- Object-oriented applications
- primitive data structures
- Java stacks, queues, & trees
- Java lists
- Java maps
- GUI user interface

- exception handlers
- graphical error messages
- file input/output

Projects:
There is a semester-long project that integrates all of the course topics.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		
Algorithms	X	
Data Structures	X	
Programming Concepts	X	
Computer Architecture		

Oral and Written Communication:
None

Social and Ethical Issues:
None

Theoretical Content:
Basic Programming

Problem Analysis:
Basic Programming

Solution Design:
Basic Programming

Prepared/Reviewed:
Prepared: Herman Harrison
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	SE 3306	**Course Title**	Mathematical Foundations of Software Engineering
Total Credits	3	**Course Coordinator**	Kendra Cooper
		URL, if any	

Current Catalog Description:

Fundamentals of logic, Boolean Logic, First-Order logic, Models of First-order logic. Completeness Theorem. Regular expressions, regular sets and finite-state machines. Fundamentals of Graph Theory, basic graph algorithms. Statecharts, Petri Nets.

Textbook(s), References and/or Other Required Material:

Discrete Mathematics and its Applications, 5^{th} edition, Rosen, McGraw-Hill, 2003

Course Goals/Objectives:

The main objective of this class is to provide a theoretical foundation in discrete mathematics for software engineers; students have the ability to apply discrete mathematics in logic, model theory, set theory, graph theory, and automata theory. In addition another objective of the class is the ability to identify, formulate, and solve the problem of formally specifying a software system using automata, Petri nets, and Statecharts. Statecharts is a formal method currently used in industry.

Major Topics Covered in the Course:

This class introduces a variety of topics in discrete mathematics that provide a theoretical foundation for software engineering. A review of propositional logic, first order logic, and naïve set theory is followed by an introduction to formal, or axiomatic, set theory and model theory for first order logic. Regular expressions, regular sets, finite state automata, finite state machines, and Chompsky's hierarchy of languages are presented. Graph theory and classic traversal algorithms are introduced. The formal methods Statecharts and Petri nets are presented and applied to the specification of software systems.

Prerequisites:

CS 2305 - Discrete Mathematics I

Projects:

The course has two formal specification assignments. In these assignments, students are provided with a description of a system written in natural language. The first assignment involves modeling a system using Statecharts. The second assignment involves modeling a system using Petri nets. Issues such as structuring the model, communication among subsystems, and concurrency are addressed.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts	X	
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course introduces the student to the basic concepts of logic and set theory.

Problem Analysis:

Addressed by assignments.

Solution Design:

None.

Prepared/Reviewed:

Prepared: Kendra Cooper
Reviewed:Gopal Gupta & Simeon Ntafos (Spring, 2005)

• Computer Science Course Description

Department Course #	CS 3305	**Course Title**	Discrete Mathematics for Computing II
Total Credits	3	**Course Coordinator**	Nancy Van Ness
		URL, if any	Responsibility of each Faculty member. e.g., www.utdallas.edu/~nancyvn

Current Catalog Description:

Topics in enumeration; principle of inclusion and exclusion. Partial orders and lattices. Algorithmic complexity; recurrence relations. Graph theory.

Textbook(s), References and/or Other Required Material:

Discrete Mathematics and Its Applications, (5^{th} ed.) by Kenneth H. Rosen, McGraw-Hill, ISBN 0-07-242434-6

References: None

Course Goals/Objectives:

The course gives the basic definitions and introduces concepts that are expanded upon in future courses. It presents much of the underlying theory in context of problem solving. Upon successful completion of this course students are expected to be able to:

- set up and solve recurrence relations,
- use the principle of inclusion and exclusion to solve problems,
- understand binary relations and their applications ,
- construct graphs and understand basic terminology,
- identify and use planar graphs and shortest path algorithms,
- understand tree terminology and use trees,
- Understand big O notation.

Prerequisites:

CS 2305 - Discrete Mathematics for Computing I

Major Topics Covered in the Course:

Recurrence relations, both homogeneous and non-homogeneous, the Principle of Inclusion and Exclusion, binary relations and their special properties, basic definitions and applications of graphs and trees as well as some algorithms and their complexity will be studied. Additional topics may include planar graphs, graph coloring, minimal graphs, shortest distances in graphs, generating functions and finite state machines as topics of study.

Projects:

Homework is regularly assigned, collected and graded. Answers are posted on the web or in other fashion made known to the students.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations	X	
Algorithms	X	
Data Structures	X	
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

While there is no formal grading of oral or written communication, students are encouraged to ask questions in class and turn in homework in a neat and readable fashion.

Social and Ethical Issues:

The only ethical issue discussed in this course is that of the integrity of the degree be kept by limiting incidences of cheating on exams, quizzes and homework.

Theoretical Content:

This course covers a lot of an introduction of theory. It has in it the theory of recurrence relations and their solutions. Basic underlying abstract concepts of graphs and trees used in more advanced courses for modeling problems are also a large part of the course. The discussion of big O and other complexity issues are first introduced. More advanced techniques of counting are also developed which will be useful in the statistics required in the degree program. Students are urged to think in more abstract ways to apply solutions in different contexts and not just adapt small changes of known algorithms.

Problem Analysis:

Much of this course is devoted to problem analysis. Students are encouraged to apply underlying concepts in various situations. While no projects are assigned the logical approach to small problems first are encouraged as means of finding a more general solution is encouraged (particularly in the advanced counting techniques). Breaking down of problems into more manageable pieces is discussed. Students are encouraged to state the problem in their own words and understand thoroughly what the question is. This is done in the context of discrete mathematical problems.

Solution Design:

This course does not deal with aspects of design

Prepared/Reviewed:

Prepared: Nancy VanNess
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS 3341	**Course Title**	Probability and Statistics in Computer Science and Software Engineering
Total Credits	3	**Course Coordinator**	Larry P. Ammann
		URL, if any	

Current Catalog Description:

Axiomatic probability theory. Calculation of probabilities of compound events, with illustrations from Computer Science and Software Engineering examples. Random variables. Synthesis of important random variables from CS/SE-related random experiments–binomial, geometric, multinomial, Poisson, exponential, and related distributions. Expectation. Important functions of random variables and evaluation of distributions of functions. Generation of random numbers of various distributions, starting from the standard uniform random number generators. Sums of independent random variables. Convolution and the use of transforms in simple cases involving exponential and Poisson random variables. Illustrative examples and simulation exercises from queuing, reliability, and program analysis disciplines. Elements of parameter (point) estimation.

Textbook(s), References and/or Other Required Material:

Probability and Statistics with reliability, Queuing and Computer Science Applications, Second Edition, K. Trivedi, John Wiley, 2002.

References:
Concepts in Probability and Stochastic Modeling, Higgins J. J and Keller-McNulty and S. Duxbury, 1994.

Course Goals/Objectives:

- Ability to understand basic properties of probability theory
- Ability to understand expectations, functions of discrete random variables
- Ability to understand concepts of conditional probability, Bayes Theorem, independence
- Ability to understand basic concepts, properties of continuous random variables
- Ability to understand expectations, functions of continuous random variables
- Ability to understand applications of the Central Limit Theorem
- Ability to understand and perform estimation of model parameters
- Ability to understand Bernoulli, Poisson, renewal processes
- Ability to understand and use Markov Chains

Prerequisites:

MATH 1236, MATH 2419 and CS 2305

Major Topics Covered in the Course:

- Basic Probability
- Counting Techniques
- Conditional Probability
- Bernoulli Trials
- Discrete Random Variables
- Discrete Probability Distributions
- Continuous Random Variables
- Statistical Inference
- Correlation and Regression
- Stochastic Processes
- Markov Chains

Projects:

The student can choose to implement the suggested project or choose his own project (approved by instructor). Project generally deals with a random phenomenon and involves the study, identification of parameters and use of simulation techniques to estimate them.

CSAB Category Content:

	CORE	**ADVANCED**
Algorithms		
Data Structures		
Theoretical Foundations	X	
Concepts of Programming Languages		
Computer Organization and Architecture		

Oral and Written Communication:

None.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The topics covered give a fundamental knowledge about the concepts of probability and statistics that are used in Computer Science.

Problem Analysis:

Addressed by the project.

Solution Design:

None.

Prepared/Reviewed:

Prepared: Pankaj Choudary
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

- ***Computer Science Course Description***

Department Course #	CS 3345	**Course Title**	Algorithm Analysis and Data Structures
Total Credits	3	**Course Coordinator**	R. N. Uma
		URL, if any	

Current Catalog Description:

Metrics for performance evaluation of algorithms. Formal treatment of basic data structures such as arrays, stacks, queues, lists, trees. Various sorting and searching techniques. Fundamental graph algorithms.

Textbook(s), References and/or Other Required Material:

Ozbirn: Data Structures & Algorithm Analysis in C++, Second Edition, by Mark Allen Weiss, ISBN 0-201-36122-1. Copyright 1999 by Addison Wesley Longman, Inc.

Wei: *Data Structures and Algorithms in C++* by Michael T. Goodrich, Roberto Tamassia and David Mount. Wiley, 2004.

Course Goals/Objectives:

The students should be able to analyze simple algorithms and solve basic recurrences. They should be able to design, implement and analyze a variety of data structures including linked lists, stacks, queues, trees, priority queues, hash tables. They should also be able to use the appropriate data structure in algorithm design for solving problems. They should know the basic sorting algorithms and their analyses. They should know and be able to apply fundamental graph algorithms.

- Analysis of algorithms including time complexity and Big-O notation.
- Analysis of stacks, queues, and trees, including B-trees.
- Heaps, hashing, and advanced sorting techniques.
- Disjoint sets and graphs.
- Course emphasizes design and implementation

Major Topics Covered in the Course:

- Big-Oh, Omega and Theta notations (2 lectures)
- Lists, Stacks, Queues (4 lectures)
- Trees (3 lectures)
- Hashing (2 lectures)
- Heaps (3 lectures)
- Sorting (3 lectures)
- Disjoint Set ADT (2 lectures)
- Graph Algorithms (5 lectures)

Prerequisites:

CS 2315 and one of CS 3305 or SE 3306

Projects:

There are about 2 written assignments and 4 programming mini-projects. The written assignments focus on the analysis part of the course. The programming projects focus on the implementation part of the course. The written assignments further their understanding of asymptotic analysis of algorithms in general and sorting algorithms in particular. They also help them to understand the application of graph algorithms. Each mini-project focuses on implementing and/or using a subset of the data structures studied. One programming project, for example, is to implement the GUI-less version of the card game *solitaire*. This enables them to work with linked lists and stacks. Another example is to implement Huffman's encoding algorithm. This enables them to work with trees and priority queues.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations	X	
Algorithms	X	
Data Structures	X	
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Introduces the student to algorithmic notations, analysis and basic data structures.

Problem Analysis:

Addressed by the assignments.

Solution Design:

None.

Prepared/Reviewed:

Prepared: R. N. Uma
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	CS/SE 3354	**Course Title**	Software Engineering
Total Credits	3	**Course Coordinator**	Douglas F. Benn
		URL, if any	

Current Catalog Description:

Introduction to software life cycle models. Software requirements engineering, formal specification and validation. Techniques for software design and testing. Cost estimation models. Issues in software quality assurance and software maintenance.

Textbook(s), References and/or Other Required Material:

Software Engineering An Object-Oriented Perspective by Eric J. Braude, John Wiley & Sons, Inc., ISBN: 0-471-32208-3

Course Goals/Objectives:

This course is intended to provide an introduction to software engineering:

1. Ability to analyze and evaluate software processes
2. Ability to establish software requirements and specifications
3. Ability to design software
4. Ability to perform verification and validation of software specifications
5. Ability to use software project management tools and techniques
6. Ability to use CASE tools for software development
7. Ability to understand formal methods in software development
8. Ability to understand functional and non-functional software specifications
9. Ability to establish and participate in an ethical software development team
10. Ability to use metrics to evaluate and forecast software developmental effort

Major Topics Covered in the Course:

- Software Lifecycles
- Software Requirements
- Software Design
- Software Specification and Validation
- Software Engineering Economics
- Software Testing
- Software Metrics
- Software Maintenance

Prerequisites:

CS 2315 or CS 3333 and CS 2305

Projects:

There is a group project. Topics are suggested by the students with the approval of the instructor.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Introduces the student to software process, lifecycle and the activities involved in various phases of the life cycle.

Problem Analysis:

Addressed by the project.

Solution Design:

Addressed by the project.

Prepared/Reviewed:

Prepared: Douglas F. Benn
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS 3375	**Course Title**	Principles of Unix
Total Credits	3	**Course Coordinator**	Martha Sanchez
		URL, if any	Responsibility of each Faculty member. e.g., www.utdallas.edu/~Martha.Sanchez

Current Catalog Description:

Design and history of the UNIX operating system. Detailed study of process and file system data structures. Shell programming in UNIX. Use of process-forking functionality of UNIX to simplify complex problems. Inter-process communication coordination. Device drivers and streams as interfaces to hardware features. TCP/IP and other UNIX inter-machine communication facilities.

Textbook(s), References and/or Other Required Material:

- *Your Unix*, The Ultimate Guide, Sumitabha Das, McGraw Hill

References:

None

Course Goals/Objectives:

After successful completion of this course, the student will be able to:

- Use Unix operating system as a user
- Use Unix programming tools as a developer
- Understand and use the concept of system calls and sockets
- Demonstrate a good level in the use of Unix Utilities
- Use the UNIX OS as server (programming and configuration)

Prerequisites:

CS 2336 Computer Science II, Or
CS 3333 Data Structures, Or
CS 3335 C and C++ or equivalent programming experience, including knowledge of C.

Major Topics Covered in the Course:

- Understanding Unix cmd
- File System and File Attributes
- Shell Fundamentals
- Shell Programming
- TCP/IP Networking Protocols
- Filters using regular expressions
- Process fundamentals

- Advanced Shell Programming
- Customizing the environment
- Perl and CGI Scripting using Perl
- Python Basics
- Unix Internals
- Topics on System and Network Administration

Projects:
There is a comprehensive project that integrates most of the course topics.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:
None

Social and Ethical Issues:
Issues related to enforcing scholastic honesty.

Theoretical Content:
Fundamentals of UNIX file systems, processes, UNIX internals and shell.

Problem Analysis:
Addressed by the project and the programming assignments.

Solution Design:
Addressed by the project and the programming assignments.

Prepared/Reviewed:
Prepared: Martha Sanchez
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	CGS 4314	**Course Title**	Intelligent Systems Analysis
Total Credits	Elective	**Course Coordinator**	Richard Golden
		URL, if any	

Current Catalog Description:

Mathematical tools for investigating the asymptotic behavior of both deterministic and stochastic nonlinear dynamical systems. Topics include: artificial neural network architectures, Lyapunov stability theory, and stochastic approximation theory. Applications to artificial neural network models of brain and behavior.

Textbook(s), References and/or Other Required Material:

Mathematical Methods for Neural Network Analysis and Design by Richard M. Golden (MIT PRESS), 1996

References:

- Anderson, J. A. An Introduction to Neural Networks. MIT Press.
- Hsu, H. Probability, Random Variables, and Stochastic Processes (Schaum's Outline)
- Marlow. Mathematics for Operations Research (Dover Book)
- Rosenlicht, M. Introduction to Analysis (Dover Book)

Course Goals/Objectives:

- Ability to select appropriate neural net design for given application problem
- Ability to read and write MATLAB code for neural nets
- Ability to read and write formal rigorous mathematical statements
- Ability to use Invariant Set Theorem to Analyze Discrete-Time Deterministic Time-Invariant Dynamical Systems
- Ability to use Invariant Set Theorem to Analyze Continuous-Time Deterministic Time-Invariant Dynamical Systems
- Ability to use Stochastic Approximation Theorem to Analyze Asymptotic Behavior of Discrete-Time Stochastic Processes
- Ability to compute gradients and Hessians of objective functions
- Ability to understand and apply basic notions of stochastic convergence
- Ability to manipulate vector-valued discrete-time stochastic processes
- Ability to manipulate matrix algebra and calculus expressions
- Ability to read and general formal statements in theorem format

Prerequisites:

- CGS 4313 or consent of instructor

Major Topics Covered in the Course:

- Introduction to Neural Networks and Mathematical Dynamical Systems. Neural Net Vector Calculus.
- Vector calculus and relevant elementary real analysis concepts for neural network analysis problems.
- Deterministic discrete and continuous time time-invariant dynamical systems.
- Types of invariant sets: equilibrium points, limit cycles, and chaotic subspaces.
- Types of stochastic convergence.
- Use of Stochastic Approximation Theorems for studying convergence of discrete-time stochastic dynamical systems with applications to artificial neural network algorithms.

Projects:

No project.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		X
Algorithms		X
Data Structures		X
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course is designed to teach the student to mathematically analyze and design artificial intelligence systems based upon artificial neural network technology

Problem Analysis:

Addressed by homework assignments

Solution Design:

Addressed by homework assignments

Prepared/Reviewed:

Prepared: Richard Golden
Reviewed:Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	CGS 4315	**Course Title**	Intelligent Systems Design
Total Credits	Elective	**Course Coordinator**	Richard Golden
		URL, if any	

Current Catalog Description:

Mathematical tools for the design and evaluation of artificially intelligent deterministic and stochastic nonlinear dynamical systems. Topics include: nonlinear optimization theory, Markov random fields, asymptotic statistical theory. Applications to theory and model construction in the behavioral and brain sciences as well as the field of artificial intelligence.

Textbook(s), References and/or Other Required Material:

Mathematical Methods for Neural Network Analysis and Design by Richard M. Golden (MIT PRESS), 1996 (ISBN=0-262-07174-6).

References:

- Anderson, J. A. An Introduction to Neural Networks. MIT Press.
- Hsu, H. Probability, Random Variables, and Stochastic Processes (Schaum's Outline)
- Marlow. Mathematics for Operations Research (Dover Book)
- Rosenlicht, M. Introduction to Analysis (Dover Book)

Course Goals/Objectives:

- Ability to select appropriate neural net design for given application problem
- Ability to read and write formal rigorous mathematical statements
- Ability to use Wolfe Conditions to Establish Convergence of Time-Varying Non-linear Optimization Algorithms
- Ability to use multivariable calculus to characterize nonlinear objective function surfaces
- Ability to use asymptotic statistical theory to make statistical inferences for non-standard neural net probability distributions on high-dimensional spaces
- Ability to verify regularity conditions for applicability of asymptotic statistical theory
- Ability to view neural nets formally as statistical pattern recognition algorithms
- Ability to use Markov Random Fields for Analysis and Design
- Ability to compute gradients and Hessians of objective functions
- Ability to understand and apply basic notions of stochastic convergence
- Ability to manipulate vector-valued discrete-time stochastic processes
- Ability to manipulate matrix algebra and calculus expressions
- Ability to read and general formal statements in theorem format

Prerequisites:

- CGS 4314 or consent of instructor

Major Topics Covered in the Course:

- Analysis of objective function surfaces. Global convergence theorem for nonlinear descent algorithms. Gradient Descent, Newton's Method, Levenberg-Marquardt, and Shanno's Algorithms.
- Applications to ANN design.
- Probabilistic knowledge representations and probabilistic reasoning.
- Markov random fields.
  Applications to the analysis and design of ANN classification objective functions.
- Constructing objective functions for learning and classification for ANN systems.
- Obtaining asymptotic distribution of the parameter estimates for hypothesis-testing and model evaluation purposes

Projects:

No project.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		X
Algorithms		X
Data Structures		X
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course will introduce concepts that help the student to use advanced mathematics to characterize the behavior of many complex simulation models and guide computer simulation research.

Problem Analysis:

Addressed by homework assignments

Solution Design:

Addressed by homework assignments

Prepared/Reviewed:

Prepared: Richard Golden
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	CS 4334	**Course Title**	Numerical Analysis
Total Credits	3	**Course Coordinator**	Janos Turi
		URL, if any	http://www.utdallas.edu/~turi/

Current Catalog Description:

Solution of linear equations, roots of polynomial equations, interpolation and approximation, numerical differentiation and integration, solution of ordinary differential equations, computer arithmetic, and error analysis.

Textbook(s), References and/or Other Required Material:

- *A first Course in Numerical Analysis,* A. Ralston and P. Rabinowitz
- *Scientific Computing with MATLAB,* A. Quarteroni and F. Saleri

Course Goals/Objectives:

- Ability to solve systems of nonlinear equations
- Ability to formulate and solve large scale optimization problems
- Ability to approximate function and data
- Ability to perform numerical differentiation and integration
- Ability to perform matrix computations
- Ability solve numerically ODEs and boundary value problems
- Ability to use MATLAB

Prerequisites:

CS 1337, MATH 2418 and MATH 2421

Major Topics Covered in the Course:

- Iterative Solution of non-linear Equations
- Interpolation Theory
- Approximation of Functions
- Numerical Differentiation and Integration
- Linear Systems
- Eigen Values and Eigen Vectors
- Numerical methods for ordinary differential equations
- Numerical methods for boundary value problems.

Projects:

No Projects.

CSAB Category Content:

	CORE	ADVANCED
Algorithms		X
Data Structures		
Theoretical Foundations	X	
Concepts of Programming Languages		
Computer Organization and Architecture		

Oral and Written Communication:

None.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course gives a basic introduction to numerical analysis.

Problem Analysis:

Addressed by the assignments.

Solution Design:

None.

Prepared/Reviewed:

Prepared: Jason Turi
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

Department Course #	CS 4337	**Course Title**	Programming Languages
Total Credits	3	**Course Coordinator**	Shyam Karrah
		URL, if any	Responsibility of Instructor

- ***Computer Science Course Description***

Current Catalog Description:

Language definition structure, data types and structures, control structures and data flow, run-time considerations. Interpretive languages; functional programming.

Textbook(s), References and/or Other Required Material:

Programming Languages – Concepts of Programming Languages, 6th Edition, by Robert Sebesta, (Addison Wesley Publication, 2003).

Course Goals/Objectives:

To gain an understanding of the fundamental concepts that underlie programming languages. To obtain a familiarity with various languages (Ada, C++, C #, Java, Smalltalk, Scheme, ML, and Prolog) which reflect these concepts. Discuss design issues of the various language constructs including programming in Functional and Logic languages.

Prerequisites:

CS 2336 or CS 3333, and CS 2305

Major Topics Covered in the Course:

- Introduction
- Syntax and Semantics
- Bindings, Scope, and Data Types
- Expressions and Subprograms
- Modular programming and Object-Oriented Programming
- Functional programming
- Logic programming

Projects:

There will be 5 to 8 assignments/programs assigned during the semester.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		
Algorithms		
Data Structures		X

Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Design considerations of basic language constructs.

Problem Analysis:

Assignments are designed to ensure that the students can apply the theoretical concepts to solve problems.

Solution Design:

Addressed by assignments which involve design and implementation of simple systems to meet the specification.

Prepared/Reviewed:

Prepared: Shyam Karrah
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS 4340 & CS 4140 Lab	**Course Title**	Computer Architecture
Total Credits	4	**Course Coordinator**	Galigekere R Dattatreya
		URL, if any	

Current Catalog Description:

Boolean algebra and logic circuits; register transfer operations; design of a small computer; input, output, and interrupt organization; powerful addressing modes, instruction formats, and their hardware structures; microprogram control.

Textbook(s), References and/or Other Required Material:

Logic and Computer Design Fundamentals (3^{rd} Ed.) by M. Morris Mano and Charles R. Kime, 2004

Course Goals/Objectives:

To develop the student's abilities to:

- Convert data between decimal and 2's complement notation.
- Perform arithmetic operations in 2's complement fixed-point fractional notation.
- Analyze and design gate-level combinational logic circuits.
- Analyze, design, and utilize combinational components such as adders, multiplexers, and decoders.
- Analyze and design simple synchronous sequential circuits.
- Design shift registers.
- Design gate-level RAM and ROM chips, utilize ROM in combinational design, and interconnect memory circuits to construct larger memories.
- Design an Arithmetic-Logic-Unit and a data path, given specific register transfer requirements and using gates and components.
- Design macros (sequences of micro-operations) for a given set of machine instructions on a simple computer, and for a given data path.

To develop the student's understanding of:

- The use of a variety of addressing modes.
- The use of priority interrupt mechanism.

Prerequisites:

CS 2305 or TE 3307

Major Topics Covered in the Course:

- Data Representation
- Boolean algebra, combinational logic gates, standard forms, and map simplification
- Analysis and design of combinational circuits
- Combinational devices

- *Combinational arithmetic circuits*
- *Sequential circuit analysis and design*
- Registers and register transfers
- ALU and data path design
- Control unit design and microprogramming
- Priority interrupt
- RAM and memory expansion
- Instruction set architecture

Projects:

No projects.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		
Algorithms		
Data Structures		X
Programming Concepts		X
Computer Architecture		X

Oral and Written Communication:

None.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Topics related to Boolean algebra and logic circuits, design of simple circuits and introduction to addressing formats.

Problem Analysis:

None.

Solution Design:

None.

Prepared/Reviewed:

Prepared: G. R. Dattatreya
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	CS 4347	**Course Title**	Database Systems
Total Credits	3	**Course Coordinator**	Joseph Leubitz
		URL, if any	Responsibility of Instructor e.g., www.utdallas.edu/~jleubitz

Current Catalog Description:

This course emphasizes the concepts and structures necessary for the design and implementation of database management systems. Topics include data models, data normalization, data description languages, query facilities, file organization, index organization, file security, data integrity, and reliability.

Textbook(s):

Fundamentals of Database Systems, R.Elmasri & S.B.Navathe, Fourth Edition, Addison-Wesley, 2003. ISBN: 0321122267.

References:

- Harvey & Paul Deitel, *Visual C++.NET: How to Program*, Prentice-Hall, 2003, ISBN: 0134373774.
- Rafe Colburn, *Using SQL (Special Edition)*, Que, 1999, ISBN: 0789719746.
- Rick Dobson, *Programming Microsoft Visual Basic .NET for Microsoft Access Database*, Microsoft Press, 2002, ISBN: 0735618194.
- Robert G. Freeman, et al., *Mastering Oracle8I*, Sybex, 2001, ISBN: 0782129293.
- Roger Jennings, *Using Microsoft Access 2003*, Que, 2003, ISBN: 0789729520.
- Roger Jennings, *Database Developer's Guide With Visual C++*, SAMS, 1995, ISBN: 0672306131.
- Gregory Speegle, *JDBC: Practical Guide for Java Programmers*, Morgan Kaufmann, 2001, ISBN: 1558607366.
- Allen G. Taylor and Virginia Andersen, *Access 2003 Power Programming with VBA*, John Wiley & Sons, 2003, ISBN: 0764525883.

Course Goals/Objectives:

- Study methods, principles, and concepts relevant to the design of database systems.
- Analyze databases from different perspectives (designer, programmer, user, administrator, etc.)
- Understand the principles and concepts relevant to the design of database systems
- Understand the different perspectives on databases (designer, programmer, user, administrator, etc.)
- Understand Data Modeling
- Understand Relational algebra and data normalization.

- Understand data organization methods, indexing, and query facilities.
- Understand data security, integrity, and concurrency.
- Understand Data Warehousing and Data Mining.
- Understand the concepts of object oriented databases and other current research.

Prerequisites:

CS 3345 – Algorithms & Data Structures

Major Topics Covered in the Course:

- Data models
- Data normalization
- Data description languages
- Query facilities
- File and index organization
- File security and data integrity

Projects:

The group project is for 3-5 students to demonstrate understanding of database technology and their ability to integrate this information to produce a working system. Analyze, specify, design, implement, document, and demonstrate a database system for the [mythical] Dallas Stock Exchange.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations	X	
Algorithms		
Data Structures		X
Software Design	X	
Programming Concepts	X	
Computer Architecture		

Oral and Written Communication:

A presentation and report is required for the group project.

Social and Ethical Issues:

Peripheral discussions of the ramifications of improper use of databases.

Theoretical Content:

Data structure, storage and indexing

Problem Analysis:

Development methodology for database systems

Solution Design:

Development methodology for database systems

Prepared/Reviewed:

Prepared: Joseph Leubitz
Reviewed: Gopal Gupta & Simeon Ntafos (Spring 05)

• Computer Science Course Description

Department Course #	CS 4348	**Course Title**	Operating Systems
Total Credits	3	**Course Coordinator**	Greg Ozbirn
		URL, if any	www.utdallas.edu/~nxm020100

Current Catalog Description:

An introduction to fundamental concepts in operating systems: their design, implementation, and usage. Topics include process management, main memory management, virtual memory, I/O and device drivers, file systems, secondary storage management, and an introduction to critical sections and deadlocks.

Textbook(s), References and/or Other Required Material:

Operating Systems, Internals and Design Principles, Fourth Edition by William Stallings, ISBN 0-13-031999-6. Copyright 2001 by Prentice-Hall, Inc.

Course Goals/Objectives:

- Understand the role of the operating system.
- Understand components of operating systems.
- Understand parallel execution in multiple processes and threads and solutions to associated problems.
- Apply principles learned in projects involving analysis, design and implementation.

Prerequisites:

CS 4340 and CS/SE 3345 or TE 3346
Working knowledge of C & UNIX

Major Topics Covered in the Course:

Processes, threads, mutual exclusion, deadlocks, scheduling algorithms, memory paging systems, file systems.

Projects:

Five or fewer projects. Some projects will address problems in concurrency which may involve multiple processes and/or threads. May also include projects in OS design and implementation.

CSAB Category Content:

	CORE	**ADVANCED**
Algorithms		X
Data Structures		X
Software Design		X
Concepts of Programming Languages		X

Computer Organization and Architecture		X

Oral and Written Communication:

Projects include written reports of design, results and conclusions.

Social and Ethical Issues:

None

Theoretical Content:

Most topics covered would be considered theory of operating systems. Many topics have their own underlying theory, for example, scheduling, memory management, and concurrency.

Problem Analysis:

Topics are presented as problems to be solved, for example, how to best manage memory and how to schedule jobs affectively. Students are also required to do projects which require problem analysis.

Solution Design:

Many solutions to operating systems problems are considered, such as in the case of memory management. Students also gain solution design experience through the projects they are required to do.

Prepared/Reviewed:

Prepared: Greg Ozbirn
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

- ***Computer Science Course Description***

Department Course #	SE 4351	**Course Title**	Requirements Engineering
Total Credits	3	**Course Coordinator**	Anthony D Sullivan
		URL, if any	

Current Catalog Description:

Introduction to system and software requirements engineering. The requirements engineering process, including requirements elicitation, specification, and validation. Essential words and types of requirements. Structural, informational, and behavioral requirements. Non-functional requirements. Scenario analysis. Conventional, object-oriented and goal-oriented methodologies.

Textbook(s), References and/or Other Required Material:

Required: *Managing Software Requirements: A Unified Approach,* Dean Leffingwell, Don Widrig, Addison Wesley: Boston

Recommended: *Requirements Engineering: Processes and Techniques,* Gerald Kotonya and Ian Sommerville, John Wiley & Sons: New York

Course Goals/Objectives:

- Ability to understand the whys, what's and how's of a software system
- Ability to differentiate process requirements from product requirements
- Ability to specify a requirements engineering process
- Ability to use a requirements engineering process specification
- Ability to identify stakeholders (and other sources of requirements)
- Ability to elicit the needs, and objectives, of stakeholders as requirements
- Ability to recognize conflicting stakeholder requirements and deal with them
- Ability to specify requirements
- Ability to validate requirements
- Ability to establish requirements traceability
- Ability to model structural requirements
- Ability to model behavioral requirements
- Ability to model non-functional requirements
- Ability to use SA and OO requirements engineering methodologies
- Ability to utilize case studies (of domain-specific) requirements engineering
- Ability to build a prototype
- Ability to use a CASE or modeling tools to capture the requirements
- Ability to outline test plans
- Ability to manage changing requirements
- Ability to understand the derivation of architectural and design models from requirements specification

- Ability to produce a clear, comprehensive and complete Software Requirements Specification

Major Topics Covered in the Course:

• Methodology for Requirements Gathering		• Prioritization
• Project Management		• Risk Management
• Project Planning		• Legal and ethical issues
• Problem Analysis		• Conflict Resolution
• Stakeholder Analysis		• CASE tool Assistance
• Elicitation Techniques		• Oral and written communication
• UML (Use Case Analysis)		• Prototyping
• Specification		• User Interface Design
• Validation		• Formal Specification
• Verification		• IEEE specifications
• Test Specification		• ISO 9000
• Conceptual Modeling		• SEI/CMM
• Model Driven Analysis		• Joint Workshop Process
• Change Management		

Prerequisites:

SE 3306 & CS/SE 3354 or consent of instructor.

Projects:

The students are divided into 3-5 person teams and given a project statement describing a business problem the solution to which will require software support. The problem is a fairly complicated process, usually related to developing a distance learning/survey process or an e-commerce web-site. They are required to organize their teams, develop a project plan to deliver a Vision document, Software Requirements Specification, and Test Plan specification. In addition they are required to conduct status briefings on a tri-weekly basis (in writing and orally) as well as a final presentation covering a skill learned (topic assigned by professor) and how it was employed in their project. They are required to use a CASE tool to maintain their artifacts

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Introduces the student to the activities involved in requirements elicitation and the various types of requirements including non-functional requirements.

Problem Analysis:

Addressed by the project.

Solution Design:

Addressed by the project.

Prepared/Reviewed:

Prepared: Anthony D Sullivan
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	SE 4352	Course Title	Software Architecture
Total Credits	3	Course Coordinator	David Russo
		URL, if any	

Current Catalog Description:

Introduction to software design with emphasis on architectural design. Models of software architecture, Architecture styles, and patterns, including explicit, event-driven, client-server, and middleware architectures. Decomposition and composition of architectural components and interactions. Use of non-functional requirements for tradeoff analysis. Component based software development, deployment and management.

Textbook(s), References and/or Other Required Material:

Software Architecture in Practice (2^{nd} Ed.) by Len Bass, Paul Clements, & Rick Kazman, Addison-Wesley, ISBN 0321154959.

Course Goals/Objectives:

- Ability to understand the Software Architectural perspective and how it differs from lower-level design
- Ability to understand the need for a Software Architecture.
- Ability to understand current era Software Architectures
- Ability to develop and apply an Software Architectural Development Fishbone Diagram
- Ability to understand and apply various Software Size and Complexity Estimation Techniques w/r/t Requirements
- Ability to develop architectural approaches from basic requirements
- Ability to analyze tradeoffs among multiple architectural alternatives
- Ability to incorporate complete (formal) requirements into a Software Architecture
- Ability to use the SEI Quality-Attribute techniques in performing Architectural Tradeoff analyses
- Ability to perform architectural reconstruction techniques using SEI scenarios
- Ability to understand Patterns and their roles in the development of software architectures
- Ability to understand the role of the MVC pattern, its limitations and abilities
- Ability to understand architectural frameworks within product line development
- Ability to communicate the necessity of architectural consistency to non-technical management.
- Ability to consistently implement an architectural specification
- Ability to construct architectures in a teamwork setting with minimal requirements

Major Topics Covered in the Course:

- The purpose of software architecture and the role of the architect
- The criteria of consistency and harmony in software architecture.

- Sizing the Software Project - FP and LOC/COCOMO
- Requirements Analysis and Development and their place in Software Architecture
- Process and Methodology and their impact on Software Architecture.
- High Level Design alternatives and tradeoffs.
- Testing Solutions
- Architecture Reconstruction
- SEI Software Architecture Quality Attributes

Prerequisites:
SE 3306 & CS/SE 3354 or consent of instructor

Major Topics Covered in the Course:
- Integration
- Scope Management
- Time Management
- Cost Management
- Quality Management
- Human Resource Management
- Communications
- Risk Management
- Conflict/Negotiations
- Procurement

Projects:
Several small projects are provided:
- Software size estimation (using student constructed LOC counters).
- Development of architecture from a product description (team based project)
- Two case studies depicting real software architectures in which the student must analyze architecture successes and failures.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:
None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Introduces the student to the various architectural styles and tradeoffs between various architectural styles.

Problem Analysis:

Addressed by the project.

Solution Design:

Addressed by the project.

Prepared/Reviewed:

Prepared: David Russo
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

- ***Computer Science Course Description***

Department Course #	CGS 4352	**Course Title**	Human Computer Interaction I
Total Credits	Elective	**Course Coordinator**	Richard Golden
		URL, if any	

Current Catalog Description:

Methods and principles of human-computer interaction (HCI), user-centered design (UCD), and usability evaluation. Provides broad overview of HCI and how HCI informs UCD processes throughout product development lifecycle.

Textbook(s), References and/or Other Required Material:

- Dix, A., Finlay, J., Abowd, G., & Beale, R. (2003). Human-computer interaction (3rd ed.). Hillsdale, NJ: Prentice Hall. ISBN 0130461091.
- Shneiderman, B. (1997). Designing the user interface: Strategies for effective human-computer interaction (3rd ed.). Reading, MA: Addison-Wesley. ISBN 0201694972
- Norman, D. (1988). The design of everyday things. New York: Basic Books. ISBN 0-385-26774-6.

Course Goals/Objectives:

- Ability to apply the principles of user-centered design, universal design, and usability assessment to create simple websites or software applications.
- Ability to describe how human information processing and cognitive limitations affect peoples' use of computer systems.
- Ability to describe the performance characteristics of humans according to the Model Human Processor (MHP) framework.
- Ability to describe the major computer input and output devices and modalities, and how they affect human performance.
- Ability to recognize and describe the main interaction models and metaphors currently in use to support human-computer interaction.
- Ability to recognize and describe next-generation models and metaphors supporting human-computer interaction.
- Ability to recognize and describe the software design process.
- Ability to describe which and how user-centered design activities should be performed during the software design lifecycle.
- Ability to describe the principles of universal access and how systems can be designed to support use by people with disabilities.
- Ability to recognize and describe the effects of workgroups and high-risk environments on the interaction of humans and computers.

Prerequisites:

None

Major Topics Covered in the Course:

- Human information processing limitations, human decision making
- Computer systems and user interfaces, human-system interaction.
- Interaction models and metaphors.
- Principles guiding well-designed human-system interaction.
- The design process - overview.
- The design process - task and user needs analysis.
- The design process – making use of task and user data for system design.
- Verifying the design – usability evaluation and testing.
- Speech User Interfaces

Projects:

The project can be either a paper or an applied project, such as: user requirements gathering (e.g., a user needs analysis for a fictional product, or a task analysis); a usability evaluation (e.g., a heuristic evaluation and usability test); a GOMS analysis on a portion of an existing product; or an approved project of your design.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		
Data Structures		X
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course introduces methods and principles of human-computer interaction, user-centered design and usability evaluation.

Problem Analysis:

Addressed by homework assignments and project.

Solution Design:

Addressed by homework assignments and project.

Prepared/Reviewed:

Prepared: Richard Golden
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CGS 4353	**Course Title**	Human Computer Interaction II
Total Credits	Elective	**Course Coordinator**	Richard Golden
		URL, if any	

Current Catalog Description:

Detailed exploration of human-computer interaction (HCI) through readings in journal articles and research reports. Practical experience in methodology typically used in the design of usable systems.

Textbook(s), References and/or Other Required Material:

- Hackos, J.T. & Redish, J.C. (1998). User and task analysis for interface design. John New York: Wiley & Sons. ISBN 0471178314.
- Norman, D. (1988). The design of everyday things. New York: Basic Books. ISBN 0-385-26774-6.

References:

Relevant articles and web-sites.

Course Goals/Objectives:

- Ability to apply the principles of user-centered design, universal design, and usability assessment to create complex websites or software applications.
- Ability to interpret the results of a keystroke-level model (KLM) analysis of a system.
- Ability to design and perform a user needs analysis.
- Ability to interpret the results of a user needs analysis and apply them to the design of a website or software application.
- Ability to represent the results of a user needs analysis in a deliverable document meant for cross-discipline consumption.
- Ability to create and document a navigation system (labeling, nomenclature, etc) for a website or application.
- Ability to create and document a site map and wireframes for a website or application.
- Ability to create and document the human-system interactions supported by a website or application.
- Ability to assess the usability of the information architecture components of a website or application (navigation system, site map and wireframes, interaction flows) via evaluative methods.
- Ability to assess the usability of the information architecture components of a website or application (navigation system, site map and wireframes, interaction flows) via test methods.

- Ability to represent the results of usability test or evaluation in a deliverable document meant for cross-discipline consumption.
- Ability to apply the results of usability assessments to improve the design of a website or application.

Prerequisites:

CGS 4352 or consent of the instructor

Major Topics Covered in the Course:

- Introductions, overview of HCI.
- Human information processing limitations, human decision making.
- Computer systems and user interfaces, human-system interaction
- Interaction models and metaphors.
- Principles guiding well-designed human-system interaction.
- The design process - overview.
- The design process - task and user needs analysis.
- The design process – making use of task and user data for system design.
- Designing for universal access.
- Verifying the design – usability evaluation and testing.
- Speech user interfaces.
- Computer-supported cooperative work; organizational and social issues.
- HCI in mission-critical and high-risk environments.
- Other interaction paradigms

Projects:

The project can be either a paper or an applied project, such as: user requirements gathering; a usability evaluation; a basic research project proposal; a GOMS analysis; or an approved project of student's own design.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		
Data Structures		X
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course introduces methods and principles of human-computer interaction, user-centered design and usability evaluation. It includes a detailed study of the design process.

Problem Analysis:

Addressed by homework assignments and project.

Solution Design:

Addressed by homework assignments and project.

Prepared/Reviewed:

Prepared: Richard Golden
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	CS 4361	**Course Title**	Computer Graphics
Total Credits	3	**Course Coordinator**	Kang Zhang
		URL, if any	

Current Catalog Description:

Review of graphic display architecture and graphic input devices. Two- and three-dimensional transformations, matrix formulations, and concatenation. Clipping and windowing. Data structures for graphics systems, segmented display files, rings, etc. Hidden line and surface elimination. Shading. Graphics packages and applications.

Textbook(s), References and/or Other Required Material:

- Leen Ammeraal, *Computer Graphics for Java Programmers*, John-Wiley & Sons, 1998.

References:

- F.S. Hill, Jr, *Computer Graphics Using Open GL*, Second Edition, Prentice-Hall, 2001.
- J.D. Foley, *et al*. *Introduction to Computer Graphics,* Addison-Wesley, 1994.

Course Goals/Objectives:

- Ability to understand the goal and applications of computer graphics
- Ability to understand and apply coordinate systems and their transformations
- Ability to understand basic 2-D drawing primitives and their implementations
- Ability to understand and apply 3-D viewing and perspective transformations
- Ability to understand hidden-face elimination problems and solutions
- Ability to implement some hidden-face elimination algorithms
- Ability to understand hidden-line elimination problems and solutions
- Ability to implement some hidden-line elimination algorithms
- Ability to understand the concepts of fractals and their applications
- Ability to develop simple user-interfaces with interactive drawing
- Ability to understand basic concepts of computer animation

Prerequisites:

- MATH 2418 – Linear Algebra
- CS 2336 - Computer Science II
- CS/SE 3345 - Data Structures and Introduction to Algorithmic Analysis

Major Topics Covered in the Course:

- Introduction to Computer Graphics
- Pixels, Lines, and Coordinate Systems

- Algorithms For Drawing Primitives
- Applied Geometry and Geometrical Transformations
- Perspective in 3-D
- Hidden-Line Elimination
- Hidden-Face Elimination
- Fractals and Self-Similarity
- Bézier Curves

Projects:

No project.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		X
Algorithms		X
Data Structures		X
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Fundamentals of graphics including pixels, lines, and coordinate Systems. Algorithms for drawing Primitives and perspective in 3D. Bézier Curves

Problem Analysis:

Addressed by homework assignments

Solution Design:

Addressed by homework assignments

Prepared/Reviewed:

Prepared: Kang Zhang
Reviewed:Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS 4365	**Course Title**	Artificial Intelligence
Total Credits	3	**Course Coordinator**	Klaus Truemper
		URL, if any	Responsibility of each Faculty member. www.utdallas.edu/~klaus

Current Catalog Description:

Basic concepts and techniques that enable computers to perform intelligent tasks. Examples are taken from areas such as natural language understanding, computer vision, machine learning, search strategies and control, logic, and theorem proving.

Textbook(s), References and/or Other Required Material:

Required: *None.*

References:

- Artificial Intelligence, Stuart Russell and Peter Norvig, Prentice Hall.
- Shadows of Forgotten Ancestors, Carl Sagan and Ann Druyan, Random House, 1992.
- The Emperor's New Mind: Concerning computers, Minds, and the Laws of Physics, Roger Penrose, Oxford university Press.
- Wittgenstein's Poker, David Edmonds, CCCO, 2001.
- Design of Logic-Based Systems, Klaus Truemper, Wiley-Interscience, 2004.

Course Goals/Objectives:

- Ability to comprehend the scope of artificial intelligence
- Ability to solve basic search problems.
- Ability to understand problems in knowledge representation and learning
- Ability to apply AI techniques to application areas of computer vision etc.

Prerequisites:

- CS 2336 – Computer Science II
- CS/SE 3345 – Data Structures and Introduction to Algorithmic Analysis

Major Topics Covered in the Course:

- Introduction and overview of Artificial Intelligence
- Search Algorithms: A*
- Knowledge Representation
- Logic Problems and Inference Methods
- Data Mining and Learning
- Intelligent Agents and systems

- Applications of AI, for instance, computer vision, natural language processing, medical diagnosis, traffic control.

Projects:

No project.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations	X	
Algorithms	X	
Data Structures		X
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Knowledge representation and learning, logic and inference methods, theorem proving

Problem Analysis:

Addressed by homework assignments

Solution Design:

Addressed by homework assignments

Prepared/Reviewed:

Prepared: Klaus Truemper
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS/SE 4367	**Course Title**	Software Testing, Verification, Validation and Quality Assurance
Total Credits	3	**Course Coordinator**	Joao W. Cangussu
		URL, if any	

Current Catalog Description:

Methods for evaluating software for correctness, and reliability including code inspections, program proofs and testing methodologies. Formal and informal proofs of correctness. Code inspections and their role in software verification. Unit and system testing techniques, testing tools and limitations of testing. Statistical testing, reliability models.

Textbook(s), References and/or Other Required Material:

Text**:** Jorgensen, Paul C., *Software Testing: A Craftsman's Approach* (2^{nd} Ed.), CRC Press, 2002, ISBN: 0849308097.

References: Lyu, Michael R., *Handbook of Software Reliability Engineering*, McGraw-Hill/IEEE Computer Society Press, 1996, ISBN: 0070394008.

Manna, Zohar, *Mathematical Theory of Computation*, Dover Publications, 2003, ISBN: 0486432386.

Musa, John D., *Software Reliability Engineering*, McGraw-Hill/Osborne Media, 1998, ISBN: 0079132715.

Selected papers from Wheeler, Brykczynski, & Meeson, *Software Inspection: An Industry Best Practice*, IEEE Computer Society Press, 1996, ISBN: 0818673400.

- Ackerman, Buchward, Lewski, "Software Inspections: An Effective Verification Process".
- Fagan, "Design and Code Inspections to Reduce Errors in Program Development".
- Fagan, "Advances in Software Inspections".

Course Goals/Objectives:

This course focuses not only on the theory of software testing but also on how testing techniques can be applied in practice to help programmers and testers function more effectively and efficiently. Special topics on the impact of testing on debugging, program comprehension, performance profiling, and reliability estimation will also be covered. In addition, projects including the use of advanced testing techniques supported by industrial tool suites are designed to help students learn the difference between state-of-art testing and state-of-practice testing.

11. Ability to understand the goal and different types of software testing
12. Ability to understand the concepts of verification and validation

13. Ability to understand and apply functional testing
14. Ability to understand and apply structural testing
15. Ability to understand and apply mutation testing
16. Ability to understand and apply GUI testing
17. Ability to understand Robustness testing
18. Ability to understand Reliability Assessment
19. Ability to understand and apply Software Testing Tools

Prerequisites:

SE 3306, CS/SE 3354

Major Topics Covered in the Course:

- Functional Testing
- Structural Testing
- GUI Testing
- Robustness Testing
- Software Verification
- Use of a Testing Tool
- Mutation Test
- Regression Testing

Projects:

Main Project: This project regards the test of a real program that has intentional errors in it. The goal is to develop a test plan that is appropriated for the specific product. The test must then be conducted according to the plan and the testing techniques/tools learned in class.

Testing Tool Software Evaluation: Students must evaluate a testing tool. The tool may be any commercially available tool or an open source, free tool. Students must prepare a paper and present it in class.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Introduces the student to the various methods of code evaluations and formal and informal proofs of correctness.

Problem Analysis:

Addressed by the project.

Solution Design:

Addressed by the project.

Prepared/Reviewed:

Prepared: Joao W. Cangussu
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

- ***Computer Science Course Description***

Department Course #	CS 4376	**Course Title**	Object-Oriented Programming Systems
Total Credits	3	**Course Coordinator**	
		URL, if any	Responsibility of each Faculty member.

Current Catalog Description:

In-depth study of the features/advantages of object-oriented approach to problem solving. Special emphasis on issues of object-oriented analysis, design, implementation, and testing. Review of basic concepts of object-oriented technology (abstraction, inheritance, and polymorphism). Object-oriented programming languages, databases, and productivity tools.

Textbook(s), References and/or Other Required Material:

- *Object Oriented Software Construction*, Bertrand Meyer, Second Edition, Prentice Hall, 2000.

References:

None

Course Goals/Objectives:

- Ability to understand the difference between the Object-Orientation and Functional Decomposition
- Ability to understand the need for a notational language for Software Design capture and communication
- Ability to understand the concept of Type and its realization as a Class (C++ and Java implementation)
- Ability to understand the elements of the Class Association Diagram – Association and Relation

- Ability to realize the aspects of Class Elicitation and Rejection (12 Principles)
- Ability to design simple simulations and realize them in an OO Language.
- Ability to understand the dynamic aspects of OO Modeling as realized in the Sequence Diagram.
- Ability to implement Object Interaction and Communication in an OO Language
- Ability to understand the concept of Object State are realized in the Harel Statechart Diagram
- Ability to implement the conceptualizations of State management using an OO Language
- Ability to understand the concept of the Thread and its correct modeling using CAD and Statecharts.
- Ability to understand the modeling of the MVC and its capture in CAD, Sequence Diagram and Statechart.
- Ability to implement MVC using an OO Language.
- Ability to implement various Visual Elements using MVC and the Interface conceptualization.
- Ability to design a simulation of an Agent based approach to solving complex problems.
- Ability to implement an Agent Based simulation design.

Prerequisites:

CS 2336 – Computer Science II or equivalent programming experience

Major Topics Covered in the Course:

- Introduction to Object-oriented methodology.
- Object-oriented Analysis and Design
- Concepts of classes and class elicitation methodology
- Inheritance and class association
- Introduction to design Patterns
- System testing fundamentals

Projects:

There is a comprehensive project that integrates all of the course topics. The project will be language neutral and the student can opt to use Java, C or C++.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations	X	
Algorithms		
Data Structures	X	
Programming Concepts		X

Computer Architecture		

Oral and Written Communication:

The students are expected to interact in the class and the project requires a final presentation.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Concepts related to object-oriented Analysis and Design.

Problem Analysis:

Addressed by the project and assignment. The students are required to develop software components based on requirements.

Solution Design:

Project requires detailed design of the problem and translation of the design solution to code.

Prepared/Reviewed:

Prepared:
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	SE 4381	**Course Title**	Software Project Management
Total Credits	3	**Course Coordinator**	Joseph Leubitz
		URL, if any	

Current Catalog Description:

Planning and managing of software development projects. Software process models, ISO 9000, SEI's Capability Maturity Model, continuous process improvement. Planning, scheduling, tracking, cost estimation, risk management, configuration management.

Textbook(s), References and/or Other Required Material:

Kathy Schwalbe, *Information Technology Project Management*, 3rd Ed., Course Technology, ISBN: 0619159847

References:

- Project Management Institute, *A Guide to the Project Management Body of Knowledge*
- Carl S. Chatfield & Timothy D. Johnson, *Microsoft Project 2002 Step by Step*, Microsoft Press, ISBN: 073561301X

Course Goals/Objectives:

20. Understand the genesis of project management and its importance to improving the success of information technology projects
21. Demonstrate knowledge of project management terms and techniques such as
    - The project management knowledge areas and process groups
    - The project life cycle
    - Tools and techniques of project management such as
        - Project selection methods
        - Work breakdown structures
        - Network diagrams, critical path analysis, and critical chain scheduling
        - Cost estimates
        - Earned value management
        - Motivation theory and team building
22. Apply project management concepts by working on a semester-long group project as team leader or active team member
23. Use Microsoft Project and other software to help plan and manage a small project

Major Topics Covered in the Course:

- Integration
- Scope Management
- Time Management
- Cost Management
- Quality Management
- Human Resource Management
- Communications
- Risk Management
- Conflict/Negotiations
- Procurement

Prerequisites:

CS/SE 3354 – Software Engineering

Projects:

The project for this course requires the utilization of project management techniques discussed in the course. The hypothetical product of the project and associated functionality are up to the project team (with instructor approval).

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		
Data Structures		X
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Introduces the student to the different activities involved in managing large software projects.

Problem Analysis:

Addressed by project.

Solution Design:

Addressed by project.

Prepared/Reviewed:

Prepared: Joseph Leubitz
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	CS 4384	**Course Title**	Automata Theory
Total Credits	3	**Course Coordinator**	DT Huynh
		URL, if any	

Current Catalog Description:

A review of the abstract notions encountered in machine computation. Topics include finite automata, regular expressions, PDAs, and context-free languages.

Textbook(s), References and/or Other Required Material:

Introduction to the theory of Computation by Michael Sipser, PWS Publishing Company, ISBN 0-534-95651-3

Course Goals/Objectives:

- Ability to design finite state automata and regular expressions
- Ability to convert among DFA, NFA, regular expressions
- Ability to show that a language is not regular
- Ability to design Push-Down Automata and Context-Free Grammars
- Ability to convert PDAs to context free grammars and vice-versa
- Ability to show that a language is not context free.

Prerequisites:

CS 3305

Major Topics Covered in the Course:

Regular Languages, Context Free Languages, Turing Machines, AND Language Decidability And Recognizability.

Projects:

No projects

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		X
Data Structures		
Programming Concepts		
Computer Architecture		

Oral and Written Communication:

None.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The concepts are aimed at introducing the fundamentals of automata theory, grammars, context-free languages etc.

Problem Analysis:

None.

Solution Design:

None.

Prepared/Reviewed:

Prepared: D. T. Huynh
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• Computer Science Course Description

Department Course #	CS 4391	Course Title	Introduction to Computer Vision
Total Credits	3	Course Coordinator	Haim Schweitzer
		URL, if any	http://www.utdallas.edu/~haim

Current Catalog Description:
Techniques for manipulating and extracting information from digital images and video. Topics include color representations, analysis and processing based on image histograms, geometric transformations, convolutions, image blurring and sharpening, extraction of edges, matching, image and video motion.

Textbook(s), References and/or Other Required Material:
None

References:
- *Computer Vision*, Ballard and Brown.
- *Computer Vision* – A Modern Approach, Forsyth and Ponce.
- *Digital Picture Processing*, Rosenfeld and Kak.
- *Digital Image Processing*, Gonzalez and Woods.

Course Goals/Objectives:
- Ability to understand the basic concepts of computer vision.
- Ability to perform basic image processing tasks
- Ability to make image transformations

Prerequisites:
CS/SE 3345 - Data Structures and Introduction to Algorithmic Analysis

Major Topics Covered in the Course:
- Color representations
- Analysis and processing based on image histograms
- Geometric transformations
- Convolutions
- Image blurring and sharpening
- Extraction of edges
- Matching
- Image and video motion.

Projects:

There will be one or more projects in which the student will be required to implement specific processing techniques.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		X
Algorithms		X
Data Structures		
Programming Concepts	X	
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

The course aims at introducing the student to the basic concepts of vision.

Problem Analysis and Design:

Addressed by the assignments and projects.

Prepared/Reviewed:

Prepared: Haim Schweitzer
Reviewed: Gopal Gupta & Simeon Ntafos

• *Computer Science Course Description*

Department Course #	CS 4392	**Course Title**	Computer Animation
Total Credits	3	**Course Coordinator**	Rafael Lacambra
		URL, if any	Responsibility of each Faculty member. www.utdallas.edu/~Rafael.Lacambra

Current Catalog Description:

Introduction to traditional animation. Kinematics of motion. Key framing. Coordinate systems and transformations (review), Euler angles and quaternions, Catmull Rom and B-Splines, Advanced Key framing, articulated figures (forward kinematics), human and animal modeling (soft tissue, skin, etc.). Facial animation (parametric). Physically based modeling (rigid, collision detection). Physically based modeling (deformable). Behavioral and heuristic models. Algorithmic animation. Optimization techniques. Animation languages and systems. Motion capture and real time control. Virtual reality and animation. Rendering and temporal aliasing. 2D and 3D morphing. 3D modeling.

Textbook(s), References and/or Other Required Material:

Computer Animation: Algorithms and Techniques, Rick Parent, Morgan Kaufmann publishers.

References:

None

Course Goals/Objectives:

- Ability to understand the role of the CS/SE major in Computer Animation
- Ability to understand different animation techniques to automate movement
- Ability to program animation techniques using software standards in industry and research
- Ability to analyze the "behind the scenes" look of animation by programming
- Ability to understand the Mathematics of computer animation

Prerequisites:

- MATH 2418 - Linear Algebra
- CS 2336 (CS2) or CS/SE 3345 - Data Structures & Algorithm Analysis

Major Topics Covered in the Course:

- Introduction to traditional animation.
- Basic Rendering Concepts.
- Modeling techniques, including splines and soft-curves
- Aids to motion specification

- Basic and Advanced Key-framing
- Articulated figure, forward and inverse Kinematics
- Physically Based
- Algorithm animation and optimization techniques
- Animation languages and systems

Projects:

There will be projects to supplement the home works. The students are expected to work on OpenGL and Maya.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations	X	
Algorithms		X
Data Structures		X
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None.

Social and Ethical Issues:

Issues concerned with enforcing scholastic honesty.

Theoretical Content:

Basic rendering concepts, modeling techniques, algorithm animation and animation languages and systems.

Problem Analysis:

The student is made to analyze the problems given as homework assignments.

Solution Design:

Addressed by the project.

Prepared/Reviewed:

Prepared: Rafael Lacambra
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• Computer Science Course Description

Department Course #	CS 4393	Course Title	Computer and Networks Security
Total Credits	3	Course Coordinator	Edwin Sha
		URL, if any	

Current Catalog Description:

The study of security and vulnerabilities in computer and network systems. Common attacking techniques such as buffer overflow, viruses, worms, etc. Security in existing systems such as UNIX, Windows, and JVM. Fundamental access control and information flow concepts. Symmetric Ciphers such as DES and AES. Public-key encryption techniques and related number theory. Message authentication, hash functions, and digital signatures. Authentication applications, IP security and Web security

Textbook(s), References and/or Other Required Material:

Cryptography and Network Security, William Stallings, Prentice Hall, Third Edition, 2003.

References:

- *Building Secure Software*, John Viega and Gary McGraw, Addison-Wesley, 2002.
- *Computer Security: Art and Science*, Matt Bishop, Addison-Wesley, 2003.
- *Hack Attacks Revealed: A Complete Reference for UNIX, Windows, and Linux with Custom Security Toolkit*, John Chirillo, Wiley, Second Edition, 2002.
- *Digital Watermarking*, I. Cox, M.Miller and J. Bloom, Morgan Kauffman, 2002.
- *Disappearing Cryptography*, Peter Wayner, Morgan Kauffman, Second Edition, 2002.
- *Computer Security*, Dieter Gollmann, John Wiley & Son Ltd., 1999
- *Cryptography: theory and Practice (Discrete Mathematics and Its Applications)*, Douglas R. Stinson, Second Edition, 2002.
- *Handbook of Applied Cryptography*, A. Menezes, P. Van Oorschot and S. Vanstone, CRC Press, 1996.
- *Applied Cryptography: Protocols, Algorithms and Source Code in C*, Bruce Schneier, Second Edition, 2002.
- *Security in Computing*, Charles Pfleeger and Shari Pfleeger, Prentice Hall, Third Edition, 2003.
- *Cryptography in C and C++*, Michael Welschenbach, Apress, 2001.
- *The Cert Guide to System and Network Security Practice*, Julia Allen, Addison-Wesley, 2001.
- *Computer Forensics*, W. Kruse II and J. Heiser, Addison-Wesley, 2002.

Course Goals/Objectives:

- Ability to identify security flaws in existing computer systems.
- Ability to identify common attacking techniques in network environments and designing common protection strategies.
- Ability to implement common encryption techniques.
- Ability to develop simple application for web security.

Prerequisites:

- CS/SE 4348 Operating System Concepts
- CS/TE 4390 Computer Networks

Major Topics Covered in the Course:

- Security in Operating Systems
- Information Security
- Common Security Attacks
- Conventional Cryptography
- Public Key Encryption
- Hash Functions and Data Integrity
- Digital Signature
- Security Practice in Email and Web
- Secure Programming

Projects:

There is a course-relevant group project and presentation.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		X
Data Structures		X
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Concepts related to system and network security, known kinds of attacks and standard protection strategies.

Problem Analysis:

None.

Solution Design:

The student can opt to design and implement a project relevant to the theme of the course.

Prepared/Reviewed:

Prepared: Edwin Sha
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS 4396	**Course Title**	Computer Networks Lab
Total Credits	3	**Course Coordinator**	Kamil Sarac
		URL, if any	Responsibility of each Faculty member. e.g., www.utdallas.edu/~~ksarac/4396

Current Catalog Description:

This course will enable students to gain hands-on experience with real networks by building networks in a laboratory environment. Projects may include establishing an intra-domain routing infrastructure in the laboratory; establishing inter-domain network topologies with BGP used to connect the different autonomous systems; running network services/applications on top of this network, including DHCP, DNS, HTTP, configuring firewalls; and network management with SNMP.

Textbook(s), References and/or Other Required Material:

- *Computer Networks*, Andrew Tanenbaum, Prentice Hall
- Mastering Networks: An Internet Lab Manual, Jorg Liebeherr and Magda El Zarki, Addison-Wesley, 2003.

References:

- *Computer Networking: A Top-Down Approach Featuring the Internet*, J. Kurose and K. Ross , Third Edition, Addison-Wesley.
- Additional material provided during the course.

Course Goals/Objectives:

- Ability to get an insight into the working of the internet.
- Ability to build and configure simple IP networks.

Prerequisites:

CS 4390 – Computer Networks.

Major Topics Covered in the Course:

- Design principles of internet protocols
- Address Resolution Protocol (ARP)
- Internet Control Message Protocol(ICMP)
- User Datagram Protocol(UDP)
- Transmission Control Protocol(TCP)
- Domain Name System(DNS)
- Routing Protocols(RIP, OSPF, BGP)
- Network-management Protocols(SNMP)

- Application level Protocols(FTP, TELNET, SMTP)

Projects:

There are lab exercises and lab reports throughout the entire course.

CSAB Category Content:

	CORE	ADVANCED
Theoretical Foundations		
Algorithms		X
Data Structures		X
Programming Concepts		X
Computer Architecture		

Oral and Written Communication:

None**.**

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

None.

Problem Analysis:

Through the lab assignments the students are taught to analyze various inter-domain and intra-domain routing problems.

Solution Design:

Students are expected to propose solutions and implement them as part of the project.

Prepared/Reviewed:

Prepared: Kamil Sarac
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• *Computer Science Course Description*

Department Course #	CS 4397	**Course Title**	Embedded Computer Systems
Total Credits	3	**Course Coordinator**	Farokh B. Bastani
		URL, if any	

Current Catalog Description:

Introduction to embedded computer applications and concepts. Real-time operating systems and resource management. Real-time scheduling and communication. Sensor data acquisition, processing and fusion. Error handling, fault tolerance, and graceful degradation. System performance analysis and optimization techniques. Includes a project to develop and analyze a small embedded computer application

Textbook(s), References and/or Other Required Material:

Required: *None*
Optional:

- *Real Time Systems*, J. W. S. Liu, Prentice Hall, 2000.
- *Real Time Systems*: Design Principles for Distributed Embedded Applications, H. Kopetz, Kluwer Academic Publishers, 1997.
- *Real-Time Concepts for Embedded Systems*, Q. Li and C. Yao, CMP Books, 2003.
- *Real-Time Systems: Scheduling, Analysis and Verification*, A. M. K. Cheng, Wiley Interscience, 2002.
- *Real-Time design Patterns: Robust Scalable Architecture for Real-Time Systems*, B. P. Douglass , Addison-Wesley, 2003.

References:

On-line references, including conference and journal papers.

Course Goals/Objectives:

- Ability to understand real-time system concepts
- Ability to analyze system performance
- Ability to use optimization techniques
- Ability to develop small to medium embedded computer application

Prerequisites:

CS 4348 – Operating Systems Concepts or equivalent

Major Topics Covered in the Course:

- Overview of Embedded applications and concepts with emphasis on the distinguishing characteristics of embedded systems and the constraints that they must satisfy.

- Distinguishing features of embedded software development process, including host/target environments and linking and memory mapping requirements.
- Brief review of the features of real-time operating systems and how they differ from general purpose operating systems.
- Real-time scheduling and schedulability analysis, including clock-driven and priority-driven scheduling.
- Specification and design methods for real-time systems, including verification using Real-Time Logic, Mode Charts and Time Petri Nets specifications.
- Resource management in real-time systems, including potential problems and their resolution as well as practical issues in building real-time systems.
- Fault-tolerance methods for embedded systems, distributed embedded systems, and real-time communication.

Projects:

There will be projects based on Wind River real-time operating system, related to process-control or communication systems.

CSAB Category Content:

	CORE	**ADVANCED**
Theoretical Foundations		X
Algorithms		
Data Structures		
Programming Concepts		X
Computer Architecture		X

Oral and Written Communication:

Projects require the students to communicate with other members of the group and are required to submit written reports for project.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty

Theoretical Content:

Theoretical information related to embedded computer systems, real-time concepts, scheduling, concurrency management etc.

Problem Analysis:

Addressed by the assigned projects.

Solution Design:

The course requires students to design and implement solutions to the projects.

Prepared/Reviewed:

Prepared: Farokh B. Bastani
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)

• ***Computer Science Course Description***

Department Course #	SE 4485	**Course Title**	Software Engineering Project
Total Credits	3	**Course Coordinator**	David Russo
		URL, if any	

Current Catalog Description:

This course is intended to complement the theory and to provide an in-depth, hands-on experience in all aspects of software engineering. The students will work in teams on projects of interest to industry and will be involved in analysis of requirements, architecture and design, implementation, testing and validation, project management, software process, software maintenance, and software re-engineering.

Textbook(s), References and/or Other Required Material:

None.

Course Goals/Objectives:

- Ability to create an appropriate software architecture for a software project
- Ability to select an appropriate target platform for a defined software project
- Ability to define a Project Plan using MS-Project and track deviation from this plan
- Ability to perform software sizing estimation using COCOMO or Function Points
- Ability to define and create a requirements document using Story Boarding techniques
- Ability to properly choose a Software Process plan and implement it within the context of available personnel
- Ability to properly choose a Software Development Methodology and implement it
- Ability to identify the subsystems within a system and annotate using package notation from the UML
- Ability to create a detailed design and hold a Design Review with the customer (instructor)
- Ability to use a Configuration Management System and develop team CM processes
- Ability to work effectively and responsibly with others in a team development environment
- Ability to use Integrated Development Environments in software development
- Ability to use CASE tools in design development and capture
- Ability to create Test Cases using Scenarios
- Ability to follow programming documentation standards
- Ability to document all design aspects of a Software Project
- Ability to track effort of development and generate cost per LOC statistics
- Ability to develop weekly progress reports and provide them to the customer
- Ability to create and use a traceability matrix between requirements and artifacts and generate statistical analysis
- Ability to present (i.e., demonstrate) a software product to the customer (instructor)

Prerequisites:

SE 4351, SE 4352, SE 4367

Major Topics Covered in the Course:

The major emphasis of this course is to allow the student an opportunity to integrate and employ all of the various components of a software engineering degree. Particular emphasis is placed on the accurate and clear engineering of a software development project. This focus is realized by requiring the student to develop a complete software architecture including, but not limited to, a project statement, platform target specification, size and development effort estimations, requirements document, high-level design, detailed design, process and methodology selection criteria, test and integration plans, requirements trace matrices.

The student concludes the effort with the requisite 'feedback' process in which they evaluate the architectural drift from the original design goals, variations in size and effort from original estimations until final deployment and a comprehensive lessons-learned document. This feedback process allows the student the opportunity to reflect and learn from the overall project effort.

Projects:

The purpose of the class is the project deliverable. The project emphasis is on the student team ability to produce a professional quality software product. Due to the time limitations of the semester and the focus on a professional quality the software product typically has to be scaled to a small size.

CSAB Category Content:

	CORE	ADVANCED
Algorithms		
Data Structures		X
Theoretical Foundations		
Concepts of Programming Languages		X
Computer Organization and Architecture		

Oral and Written Communication:

None.

Social and Ethical Issues:

Issues related to enforcing scholastic honesty.

Theoretical Content:

Project course. Focus is on providing the student hands-on experience in all aspects of software development.

Problem Analysis:

Addressed by the project.

Solution Design:

Addressed by the project.

Prepared/Reviewed:

Prepared: David Russo
Reviewed: Gopal Gupta & Simeon Ntafos (Spring, 2005)