Course Syllabus

Course Information
(course number, course title, term, any specific section title)

Course Prefix, Number, Section
CS3376.001, SE3376.001
CS 3376.0W1, SE 3376.0W1

Course Title
CS/SE 3376.0W1 – C/C++ Programming in a UNIX Environment

Term
Fall 2016

Professor Contact Information
(Professor's name, phone number, email, office location, office hours, other information)
Richard Min
Office Telephone: 972-883-4522
Email: rkm010300@utdallas.edu
Office: ECSS 4.609
Office Hours: Tuesday & Thursday 2pm-5pm, Monday & Wednesday 7pm-9pm

---

Course Pre-requisites, Co-requisites, and/or Other Restrictions
(including required prior knowledge or skills)
Prerequisite: ECS 2336 or equivalent. (3-0) S
[Computer Science I and II (CS 1337 and CS 2336 or equivalent courses) and Basic Object Oriented Programming Skills with C++]

Course Description
CS 3376 C/C++ Programming in a UNIX Environment (3 semester hours) Advanced programming techniques utilizing procedural and object oriented programming in a UNIX environment. Topics include file input and output, implementation of strings, stacks, queues, lists, and trees, and dynamic memory allocation/management. Design and implementation of a comprehensive programming project is required.

Student Learning Objectives/Outcomes
1. Ability to use the UNIX operating system interactively as a user (commands)
2. Ability to express algorithmic solutions using shell scripting (utilities)
3. Ability to understand and use regular expressions
4. Ability to use the UNIX programming environment (editor, compiler and linker)
5. Ability to understand UNIX processes (creation and control)
6. Ability to perform input/output of binary files
7. Ability to use interprocess communication (pipes, sockets and signals)
8. Ability to understand the UNIX file system
9. Ability to understand and use version control system

Required Textbooks and Materials

Required Textbooks

1. *Advanced Programming in the UNIX® Environment*, 3e. W. Richard Stevens and Stephen A. Rago. Addison-Wesley. © 2013. ISBN-10: 0-321-63773-9. ISBN-13: 9780321637734
(Available online & free via UTD Library => eBook => Safari)

2. *A Practical Guide to Linux® Commands, Editors, and Shell Programming*, Third Edition. Mark G. Sobell. Prentice Hall. © 2012. ISBN-10: 0-13-308504-X. ISBN-13: 9780133085044
(Available online & free via UTD Library => eBook => Safari)

3. Gaddis, Starting Out with C++ From Control Structures through Objects (with Access) 8th edition. ISBN-10: 0133796337 • ISBN-13: 9780133796339. (7th edition is OK, 0132576252)
(This is the textbook for your cs1336 and cs1337 courses. We will review chapters 12-19 mostly through ppts).

Required Materials

1. Unix® and Linux® System Administration Handbook, Fourth Edition, Video Enhanced Edition. by Evi Nemeth; Garth Snyder; Trent R. Hein; Ben Whaley. © 2010 Prentice Hall. ISBN-10: 0-13-148005-7. ISBN-13: 978-0-13-148005-6
(Available online & free via UTD Library => eBook => Safari)

2. Unix Systems Programming: Communication, Concurrency, and Threads. Kay A. Robbins; Steven Robbins. © **2003** Prentice Hall. **ISBN-10:** 0-13-042411-0. **ISBN-13:** 978-0-13-042411-2
(Available online & free via UTD Library => eBook => Safari)

3. C for Programmers with an Introduction to C11. by Harvey Deitel and Paul Deitel. © 2013 Prentice Hall ISBN-10: 0-13-346206-4. ISBN-13: 978-0-13-346206-7
(Available online & free via UTD Library => eBook => Safari)

4. 21st Century C, 2ed. Ben Klemens. © 2014 O'Reilly Media, Inc. ISBN-13: 978-1-4919-0389-6
(Available online & free via UTD Library => eBook => Safari)

5. Intermediate C Programming. Yung-Hsiang Lu. © 2015 CRC Press. ISBN 978-1-4987-1163-0.
(Available online & free via UTD Library => eBook => Safari)

6. C++ How to Program, 10/e. by Paul Deitel and Harvey Deitel. © 2016 Pearson. ISBN-13: 978-0-13-444823-7
ISBN-10: 0-13-444823-5
(Available online & free via UTD Library => eBook => Safari)

7. *C++ Programming Language.* 4/e. Stroustrup ©2014 Addison-Wesley ISBN-10: 0321958322. ISBN-13: 9780321992789
(Available online & free via UTD Library => eBook => Safari)

8. Unix and Linux: Visual Quickstart Guide, Fifth Edition. by Eric J. Ray; Deborah S. Ray © 2014 Peachpit Press
ISBN-10: 0-321-99754-9. ISBN-13: 978-0-321-99754-8
(Available online & free via UTD Library => eBook => Safari)

Online Resource and Web Sites

C++ language tutorial http://www.cplusplus.com/files/tutorial.pdf

C++ tutorial http://www.learncpp.com/
C++ reference: http://cppreference.com
MobaXterm: http://mobaxterm.mobatek.net/
Unix/Linux commands: https://kb.iu.edu/d/afsk
Linux Shell and Commands: http://vic.gedris.org/Manual-ShellIntro/1.2/ShellIntro.pdf

---

Suggested Course Materials
See Required Materials

Assignments & Academic Calendar
(Topics, Reading Assignments, Due Dates, Exam Dates)

60% for 3 Tests. 20% for each 2-hour test. Tentatively scheduled: Friday 3pm–9pm on (1) 9/30, (2) 11/04, (3) 12/02. Each test will be taken at Testing Center (Student Assessment Center, McDermott Library 1st floor) for 2-hour examination. Time of Test will be announced later in elearning. Each student should make a seat reservation prior to each test. All exams are closed book and closed notes. Exams will focus more on concepts and less on details. Necessary documentation will be provided to avoid the need for memorization as much as possible. We will likely take all the tests in the testing center as scheduled. You can expect to see a few coding/analysis questions, a few short answer questions and a few multiple-choice questions in each test. Instructor is responsible for grading all the tests. Any make-up tests will be scheduled during the same week (usually Tuesdays prior to the actual test date) at the discretion of the instructor. There should be a valid reason for scheduling make-up tests & they need to be coordinated with the instructor, 1-2 weeks prior to the test date except for serious medical condition (with Doctor's or Hospital's certificate will be required as a valid proof.) It is unlikely that curving will be used to boost the final grades. If the instructor decides to do it, only the test scores will be boosted, but the tests' contribution will be clipped at 60%. In other words, curving will NOT make up for the points lost in all other assignments. So, it is extremely important to complete them in timely manner.

20% for 4 Assignments (projects) contributing 5% each. Due (Monday 12pm Noon): (1) 9/19, (2) 10/17, (3) 11/14, (4) 12/05. You can ask for clarifications and help in the weekly forum. If you need help with your code, it is ok to post 1 or 2 lines of code, but do not post your full program - email it to TA or professor instead. You are expected to start working on them as soon as they are posted. Do not expect us to rescue you on the day of submission. I encourage everyone to submit the projects 1 or 2 days early. You can upload it again but the last submission will be graded. [Do not wait until the last minute to submit it. I do understand things happen and occasionally as you may not be able to submit projects on time.] The Late Penalty policy is to assess 1% penalty for every 1 hours. For example, if you submit the projects exactly 1 day later, 24% penalty will be assessed. Late projects will be accepted up to 3 days and thereafter 0. You won't be able to submit it after 3 days and your project grade will be set to 0. My advice is to submit whatever you have done (your best effort) before/by the due, to seek for any further discretion and/or consideration. All these assignments/projects should be done in Linux and you will hand-in your projects directly in Linux. We will NOT use elearning to submit the projects, but your grades and TA's comments will be recorded there - you can click on My Grades to access them. More details on Assignment & Submission steps will be given with eLearning.

20% for Weekly Activity & Quiz (including online quiz) will be posted by Monday & will be due Saturday midnight (11:59pm) every week. It will be a small programming exercise or tryout (e.g., to write and run a simple "Hello world" program, to try Linux commands or sample programs provided, to install a tool to try it) in most weeks. It can also be a quiz (online and open-book) or some other meaningful activity as well. It will vary every week. Each weekly activity and its score may vary case by case. Late submissions are NOT accepted for weekly activities and quizzes. Note: Weekly quiz will provide a good snapshot, an excellent opportunity to review, and for a preparation for each test. Late submissions are NOT accepted for weekly activity or quiz. **Weekly Postings. 2** meaningful and relevant posts are required every week in weekly discussion forums. This is extremely crucial component of a true online course. No non-

sense and no trivial comment. One-liners saying "Thanks!" ("Weather is bad" or "I got it" or "I do not know" or "very good" etc.) will not be counted as a valid posting or participation. Keep your posting very relevant and valuable to you and your classmates, and to the course work and activity of the week. Your post can be a good question, meaningful response to another student's question, interesting observation, etc. For a question, you should do your own homework for your question and share your findings. If you use an external source, you should provide a reference or a link of the source, and provide a good overview or summary in your wording. Do not post any offending or destructive content. Do not post any overwhelming contents (e.g., to copy and paste big image or images, or very long text content, or using "big" fonts) but you should attach a file as you need. In simple words, each post should value to the course. Instructor (TA or Grader) will grade the weekly forum and determine the value of each post - instructor's decision is final. First post should be submitted latest by Wednesday midnight and 2nd post should be completed latest by Saturday midnight, otherwise respective posts won't receive any grade. It is possible for someone to be a silent observer in on-ground course and still manage to get the final grade of A. It is impossible to do it in online course. Reasonable progress towards the expected answer or learning will get 1 point & perfect or near-perfect submissions will get 2 points. Late submissions are NOT accepted for weekly posts.

The following Table is for Weekly Activity (See the detail for elearning).

UNIT/ DATES	TOPIC/LECTURE	READING	ASSESSMENT / ACTIVITY	DUE DATE
0	Orientation & Prerequisite Form		Week00 Activity (to sign and upload the completed prerequisite form)	8/31
1 8/22-8/27	Scroll down to see the syllabus! UNIX, Linux and C overview Lecture (see it in 720p setting - low quality video - first 2 minutes - not sure why) intro-to-linux.pdf inode_structure.pdf Linux directory structure	Review Gaddis1-5 (ppts) Read APUE1-2	Week01 Activity (to download, install and try putty, etc.) See Week01 Activity folder for detail	8/27
2 8/28-9/03	Working with Shell - Lecture - Part1 Part2 redirections.ppt Working with Shell ... ln_cp(1).ppt Lecture Part1 Part2 Command line operators Web resources: google VI commands emacs commands	Review Gaddis6-7 (ppts) Read APUE3	Week02 Activity	9/03
3 9/04-9/10	Linux Commands & pipes (continued): Lecture: Part1 Part2 intro-to-linux_final.pdf Log processing using pipes: Video Linux commands: grep, chmod and xargs Back-quote operator `...`: back_quote.swf C/C++ fundamentals: Lecture: Part1 Part2	Review Gaddis9-11 (ppts) Read APUE4	Week03 Activity	9/10

4 9/11-9/17	C/C++ fundamentals ... basicsCC++(1).pdf C programs: Part1 Part2 Namespaces: pages 120-122 of C++ tutorial Strings: lecture, Chapter10.pdf Working with Files, Linux Environment iostreams.pdf Lecture: Part1 Part2	Review Gaddis12 (ppts) Read APUE5	Week04 Activity	9/17
5 9/18-9/24	C/C++ Classes and Objects, Arrays Lecture (ComplexType class): Part1 Part2 Lecture (friends, Product, Course): Part1 Part2 classes_arrays(2).pdf	Review Gaddis13 (ppts) Read APUE6	Week05 Activity	9/24
6 9/25-10/01	C/C++ Classes and Objects, Arrays ... course_class.pdf Pointers.pdf Lectures: Part1 Part2 Part3	Review Gaddis14 (ppts) Read APUE7	Week06 Activity	10/01
7 10/02-10/08	More Classes & OOP: more_classes.pdf Lecture Inheritance & Virtual methods Lecture inheritance.pdf	Review Gaddis15 (ppts) Read APUE8	Week07 Activity	10/08
8 10/09-10/15	Exceptions & Templates Lecture Exceptions_Templates.pdf Lecture #2 Video: gdb intro: intro_gdb.swf	Review Gaddis16 (ppts) Read APUE9	Week08 Activity	10/15
9 10/16-10/22	Advanced File and I/O operations & Recursion serialization.pdf Lecture Inheritance 2nd half:Lecture inheritance_final.pdf	Read APUE10 (Signal)	Week09 Activity	10/22
10 10/23-10/29	POSIX Threads, Semaphores processes-threads4.pdf Lecture processes-threads(1).pdf Lecture Processes and Signals lecture processes-threads.pdf	Read APUE11-12 (Thread)	Week10 Activity	10/29

11 10/30-11/05	pipes.pdf lecture pipe_code.txt pipe example: lecture pipes_diagrams.pdf	Read APUE15 (IPC & Pipe)	Week11 Activity	11/05
12 11/06-11/12	Development Tools - Makefiles, Debugging make-debug.pdf lecture	Read APUE16 (Socket)	Week12 Activity	11/12
13 11/13-11/19	Review: Pointer, Structure, IO (Gaddis ch09, ch11, ch12) Advanced Topic (TBA)	Read APUE13 (Adv IO)	Week13 Activity	11/19
14 11/20-11/26	Break			
15 11/27-12/03	Pointers & dynamic memory allocation Lecture: Part1 Part2 PDF Programs: Part1 Part2 pointer_diagrams.pdf Dynamic List: lecture Advanced Topic	Review Gaddis17 (ppts)	Week15 Activity	12/03
16 12/04-12/10	Linked lists, Stacks & Queues, Binary Trees Linked Lists Stacks Queues.pdf Lecture Store variations: Lecture **Advanced Topic & Review**		Week16 Activity	12/10

Grading Policy
(including percentages for assignments, grade scale, etc.)

Letter grades will be assigned as follows:

97-100	A+	93-96	A	90-92	A-
87-89	B+	83-86	B	80-82	B-
77-79	C+	73-76	C	70-72	C-
67-69	D+	63-66	D	60-62	D-
Below 60	F				

Note: "Running" and weighted total in your gradebook shows the current weighted grade based on your graded work only based on what you have submitted. For example, if you have done only Test1, Assignment1, Weekly postings so far (but you have missed Test2 and missed Assignment2 totally), current total grade will be based on only those entries that you have submitted and done.

Course & Instructor Policies

(make-up exams, extra credit, late work, special assignments, class attendance, classroom citizenship, etc.)

Instructor is responsible for grading all the tests & weekly participation. TA will be responsible for grading projects and weekly assignments. So, contact the TA directly for any grading related discrepancies for programs. It is not possible to give a detailed feedback for each project/weekly assignment/test question due to large # of students in our classes. If you need more details/clarification, you are encouraged to meet the TA/instructor during office hours & get personal attention. Do not rely on email alone to get the full response. If you are stuck with your assignment, it is better to turn in what you have and send us email. We will revise your submission and give some guidance. Your next submission will override the previous submission - TA will always grade the latest submission for each project. You can use email to get help for weekly assignments. Include the detailed problem description & applicable error messages, zip all your source files and include it with your email too. Do not just say "my program does not work" and expect us to figure out everything - you need to help us to help you efficiently. We expect to complete grading assignments (projects), weekly activities or quizzes, and tests in a week or so. However, when the schedule gets too busy, it can be as long as 2 weeks before the grades are assigned. It is the students' responsibility to review the grade details when they become available and follow up for clarifications if needed. For in-class course (and elearning weekly activity & participation via elearning for online course), Attendance Rule & Policy: Please note that if you miss any lectures beyond the 1st week, then automatic actions kick in: (1) Missing the next lecture in the 2nd week will result in an automatic drop of one grade from your final course grade. (2) Missing the entire 2nd week of lecture(s) is an automatic F in the course. So if you are going to miss more than one week of classes (ideally, you should not miss any lecture, but sometimes people switch courses during the first week), then you should not be in the course and you should drop out. Further you should plan to be here for Final Examination Week, as it will be scheduled for this course.

Off-campus Instruction and Course Activities

Below is a description of any travel and/or risk-related activity associated with this course.

Each student should plan to take 3 Tests at Testing Center, and demo for Assignments, or to arrange off-site proctoring service for Tests with Testing Center.

Comet Creed

This creed was voted on by the UT Dallas student body in 2014. It is a standard that Comets choose to live by and encourage others to do the same:

"As a Comet, I pledge honesty, integrity, and service in all that I do."

UT Dallas Syllabus Policies and Procedures

The information contained in the following link constitutes the University's policies and procedures segment of the course syllabus.

Please go to http://go.utdallas.edu/syllabus-policies for these policies.

The descriptions and timelines contained in this syllabus are subject to change at the discretion of the Professor.