CS1337.009 Course Syllabus

Spring Semester, 2017

Course Number:	CS 1337.006
Course Title:	Computer Science I
Room:	GR 4.301
Meeting Time:	Mon & Wed from 8:30 to 9:45 AM
Credit Hours:	3
Instructor:	Dr. Jey Veerasamy Office: ECSS 3.231 E-Mail: jeyv@utdallas.edu Web Site: utdallas.edu/~jeyv
Office Hours:	Mon/Tue/Wed/Thu 10 – 11am By appointment.
Teaching Assistant:	TBA
Prerequisites:	CS 1336 or equivalent programming experience. (3-0) S

Catalog Description:

Review of control structures and data types with emphasis on structured data types. Applies the object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design. Includes basic analysis of algorithms, searching and sorting techniques, and an introduction to software engineering. Programming language of choice is C/C++.

Course Expectations:

After successful completion of this course, the student should have an:

1. Ability to use single and multi-dimensional arrays
2. Ability to implement linear and binary searches
3. Ability to implement simple sorting algorithms
4. Ability to implement structured data types
5. Ability to define and implement a class

6. Ability to use fundamentals of object-oriented design

Tentative schedule: Schedule will be updated along the way within elearning.

Week	Topic	Reading
1	Syllabus review, course requirements, terminology.	Ch. 1
	Introduction to C++; Binary arithmetic review	Ch. 2
2	MLK day – holiday!	
	Expressions and Interactivity; Making Decisions	Ch. 3, 4
3	Loops and Files; Functions	Ch. 5, 6
	Arrays	Ch. 7
4	Arrays; Searching and Sorting Arrays	Ch. 8
	Pointers	Ch. 9
5	Pointers, continued	
	Characters, C-Strings, and the String class	Ch. 10
6	Structured Data	Ch. 11.1-11.10
	Structured Data, continued	
7	Introduction to Classes	Ch. 13
	Introduction to Classes, continued	
8	More About Classes	Ch. 14
	More About Classes, continued	
9	Test #1 Review	Chapters 1-14
	Test #1	
10	Spring Break	
	Spring Break	
11	Inheritance, Polymorphism, and Pure Virtual Functions	Ch. 15
	Inheritance, Polymorphism, and Pure Virtual Functions	
12	Inheritance, Polymorphism, and Pure Virtual Functions	
	Recursion	Ch. 19
13	Recursion	

	The Standard Template Library; Error Handling	Ch. 16
14	The Standard Template Library; Error Handling	
	Advanced File I/O	Ch. 12
15	Advanced File I/O	
	Exam Review	Chapters 12, 15, 16, 19
16	**Test 2**	

Textbook:

Starting Out with C++, From Control Structures through Objects (8th edition); Gaddis, Tony; Addison-Wesley Publishing. ISBN 978-0-13-376939-5 (7th edition is OK as well)

Notes regarding textbook material:
As you read the text watch the corresponding VideoNotes. The VideoNotes are available at http://www.pearsonhighered.com/gaddis/.

Additional optional resources:
Programming
using JavaScript: www.khanacademy.org/cs, www.utdallas.edu/~jeyv/kajs
C++ language tutorial www.cplusplus.com/files/tutorial.pdf
C++ reference: www.cppreference.com
C++ tutorial www.learncpp.com

Course & Instructor Policies:
The final grade will be computed as follows:

Tests	50%	*2 tests contributing 25% each* All tests are closed book and closed notes & they will be conducted in the computers @ the testing center. If the testing center is not available, paper test will be given. Tests will focus more on concepts, but coding will be required for almost every question. Tests will NOT have multiple-choice questions. Necessary documentation will be provided to avoid the need for memorization as

		much as possible. All make-up tests are scheduled during the week following the actual test date at the discretion of the instructor - advance notice is required.
Assignments	50%	*There will be an assignment due every week. All of them will have equal weightage.* I encourage everyone to submit the assignments 1 or 2 days early. Do not wait until the last minute to submit it. But I do understand things happen and occasionally you may not be able to submit assignments on time. My policy is to assess 1% penalty for every 2 hours. For example, if you submit the assignment exactly 1 day later, 12% penalty will be assessed. Late assignments will be accepted up to 4 days. You won't be able to submit it after 4 days and your assignment grade will be set to 0. You can ask for clarifications and help in the Assignments 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.

Course credit is only given for work assigned in the course schedule. No extra work will be assigned nor will extra credit be given for any extra work performed by a student. Instructor is responsible for grading all the tests, quizzes & weekly participation. TA will be responsible for grading assignments and weekly activities. So, contact the TA directly for any grading issues related to those items. If you cannot resolve it with TA, bring it to instructor's attention.

In addition to meeting the instructor before or after the class, you can also visit the instructor or TA during respective office hours. You can call instructor's office phone during office hours as well. However, be prepared to hold and wait if the instructor is busy with another student in the office. Additionally, you are welcome to email the instructor or TA. If you need help with your program, in addition to problem description & applicable error messages, include your source files with your email too, so that we can review & help you efficiently.

Letter grades will be assigned as follows:

97-100 A+	94-97 A	90-94 A-
87-90 B+	84-87 B	80-84 B-
77-80 C+	74-77 C	70-74 C-
67-70 D+	64-67 D	60-64 D-
Below 60 F		

Programming Assignments:

Complexity level of each assignment may vary – students will be expected to spend a few hours in a computer every week. Right way to approach the programming assignments is to start on them at least one week earlier than the due date & get help when you get stuck (you can approach the instructor, TA, or tutors at CS tutoring lab for help). Do not waste lots of hours trying to fix a small glitch. In simple words, your approach will determine whether programming assignments provide an enjoyable learning experience or end up like a painful useless activity.

Programming assignments will be graded on a 100 point basis, utilizing the following criteria:

		Max Score
Source Code	Overall design	40
	Formatting	10
	Naming	10
	Capitalization	10
Execution	Typical cases	25
	Special cases	5
Total		100

Programming assignments should be turned in by means of eLearning. You need to submit only .cpp files for individual assignments, unless explicitly stated otherwise.

Any standard C++ compiler and Integrated Development Environment (IDE) can be used to develop, debug and run your programs. Microsoft Visual Studio, Microsoft Visual Express, Code::Blocks, NetBeans, Eclipse and jGRASP are a few popular tools. More information about these tools will be provided in elearning.

Course attendance policy

Attendance policy for this course is bit stricter than CS dept policy. Three absences will result in one letter grade drop. Four absences will result in F. Attendance will be taken in every class and the absence count will be updated in the grade book after every class.

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.