

CS1337.0U2 – Computer Science I

Summer 2017

Syllabus

Contact Information

Professor: Khiem Le, Ph.D.
Office: ECSS 3.703
Office phone: 972-883-6217
Email: kvl140030@utdallas.edu (best way to reach me)
Lecture: Tuesdays and Thursdays, 3:00 to 5:15 PM @ECSS 2.312
Office hours: Tuesdays and Thursdays, 5:30 to 6:30 PM, or by appointment

Grader: TBD
Email: TBD
Office hours: TBD
Location: TBD

Note: Email is the best way to reach me, and please include in your email subject title the class number and section. Remember to sign your email, so I know where it is coming from. That will help me respond to you sooner.

Course Description

Computer Science I (3 semester credit hours) 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++.

Prerequisites and Corequisites

Prerequisite: [CS 1336](#) with a grade of C or better or equivalent.

Required Textbook and Material

Textbook

Starting Out With C++, From Control Structures through Objects, Eighth Edition, by Tony Gaddis, Addison Wesley, 2015.

Additional course materials, such as assignments, sample programs, and other materials will be provided as needed through eLearning at <http://elearning.utdallas.edu>.

Student Learning Objectives

The Learning Objectives of this class are as follows:

1. Ability to use single and multi-dimension 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

Grading

Your letter grade will be determined from an overall numerical score, calculated as a weighted average with the weights below:

Test #1 score:	25%
Test #2 score:	25%
Test #3 score:	25%
Homeworks average:	20%
Quizzes average:	5%

The overall numerical score is possibly curved and then converted to a letter score, as follows. Curving, if any, will always be in your favor. For example, if you have an overall numerical score of 91 before curving, you are guaranteed to get at least A-.

In addition, if you are borderline, at my discretion, I may decide, at my discretion, to bump you up based on the following criteria:

- Class attendance and citizenship
- Your going to the Computer Science Mentoring Center (CSMC) to get help when needed. If you go there, make sure your going to the CSMC is recorded
- Improvement throughout the semester

Overall numerical score (possibly curved)	Grade
≥97	A+
≥93	A
≥90	A-
≥87	B+
≥83	B
≥80	B-
≥77	C+
≥73	C
≥70	C-
≥67	D+
≥63	D
≥60	D-
Less than 60	F

Important notes:

- **In order to receive a grade higher than a C-, you must have a homeworks average of at least 70%.**
- **According to the CS department attendance policy, if you have three consecutive unexcused absences, your grade will be automatically downgraded by one letter grade. For example, an A- would be downgraded to a B-. If you have 4 consecutive unexcused absences, your grade will automatically be an F.**

Tests

- Each test covers all the handouts + information given during lectures + homeworks + class discussions + exos + quizzes, up to the exam. This means test #2 is a comprehensive exam
- Electronic devices (e.g. computers, laptops, cell phones, tablets) and backpacks will not be allowed at desks during tests
- Closed book, closed notes
- Tests are taken on eLearning. Types of questions that may be found in tests are essay, true/false and multiple choice questions. By essay, I mean any question for which you do not answer simply by checking a box
- Each test is graded out of 100
- **You are required to take the tests on the regular date. Exceptions to this policy are only made in very rare circumstances, typically due to unforeseen circumstances such as a medical or family emergency. All makeup exams are scheduled and given at the discretion of the instructor. They are only given to students who contact the instructor prior to the originally scheduled exam date/time, or for a justified emergency with documentation.**

Homeworks

- Homeworks are programming projects designed to supplement our class discussions and the textbook, and give you an opportunity to practice the concepts learned. You will usually have one week to do each one, but in specific instances, the time allocated may be different than one week. You are notified of the due date when a homework is assigned.

- All homework assignments will be submitted to eLearning. The TAs will download them, grade them, and upload the resulting grade with comments in the “Feedback to Learner”. Read the “Feedback to Learner” to learn about what mistakes, if any, you made so you can learn from your mistakes
- Each individual homework assignment will be graded out of 100.
- The homeworks average is the average of the individual homework scores.

Submission Policies

- An assignment that is turned in late, but is still within the first 24 hours after the due date, will receive a 20% penalty on the grade. That is, the homework is graded normally, and the score is multiplied by 0.80 to yield the actual score for that homework. Assignments more than 24 hours late are not accepted.
- For some specific homeworks, it could be that late submissions will not be accepted. If and when that happens, you will be notified when the homework is assigned.
- All submissions must be your individual work. If you get help from others (other students, CSMC) you must ensure that you submit only work that you have personally done. **Non observance of these rules may be considered as academic dishonesty and handled accordingly.** There are no group assignments in this class.

Quizzes

- Quizzes usually take place about a week after a section or chapter has been completed.
- They are designed to give you a gauge as to how well you grasped the material, and prepare you for the tests and homeworks.
- Quizzes are taken in class and could be true/false, multiple choice or essay questions. The questions relate to the key points of the section or chapter that has been covered
- There is no makeup quiz. If you miss a quiz due to an unexcused absence, you will get no credit
- The maximum achievable score on each individual quiz may vary from quiz to quiz, as it depends on the number of questions in the quiz
- The quiz average is the weighted average of the quiz scores, where the weight of a quiz is proportional to the maximum achievable score of the quiz. The average is normalized to be a score out of 100. For example, assume there are 3 quizzes, and quiz-1, quiz-2 and quiz-3 have 10, 20 and 30 questions respectively. If each question is 5 points, the maximum achievable scores of quiz-1, quiz-2 and quiz-3 are 50, 100 and 150 respectively. The quizzes average will be $(s_1+s_2+s_3)*100/(50+100+150)$, where s_1 , s_2 and s_3 are your scores on quiz-1, quiz-2 and quiz-3 respectively.

In-class Exos

I often teach the programming concepts by illustrating them with a live program that I type, compile and run as a demonstration in class. You will be asked to type, compile and run the same program along with me in class, and to submit your program on eLearning before the end of the lecture. Your program will not be graded, the main purpose of the exos are to engage you in the learning and take your attendance. It is not critical that your program compiles. It is more important that you pay attention and listen than trying to make your program work. If the classroom is not equipped with a PC for each student, I strongly recommend you bring your own

personal laptop. Another means to take your attendance will be used if you cannot bring your laptop.

Class Attendance and Citizenship

- Class attendance
 - Students who regularly attend class tend to make significantly higher grades than those who do not.
 - Attendance record is based on quiz participation and in-class exo submissions. At my discretion, other means to take attendance may be used.
- Citizenship
 - Good citizenship, which is behavior demonstrating effort to learn and respect of other students' effort to learn
 - You are encouraged to participate in class discussions and ask questions, whether in class or out
 - Disruptive behavior in class is not tolerated.
 - You are expected to be on time and stay till the end of the lecture. If you ever need to leave early or come late, you must minimize disruption to the lecture.
- **Class attendance and behavior will be a consideration for possibly bumping you up if you are a borderline case.**
- Some absences are automatically excused by the school and won't count against you. These include absences for sporting events (if you're a member of a UTD sports team) and other situations. If any of these apply to you, you have to contact me **beforehand** and we'll make arrangements for it. In addition, absences for medical reasons will be excused with documentation.

Issues about Grading

Grade Dispute: Students are required to bring up any grading issue within a week of grade posting.

- Contact the grader for questions about the homework and quiz scores. **Please copy me on all your emails with the grader so I am aware of the situation and can make sure it is resolved.**
- Contact me for questions about the test scores.

Course Tools

Communication

Assignments, grades and announcements are posted on eLearning. Announcements are also emailed out to the whole class. In addition, you may also receive individual emails from me or the grader. **It is your responsibility to logon to eLearning and check your UTD email to stay abreast of assignments, announcements and other information.**

Programming Tools

All of the programs we write this semester will be in C++, and we will be using C++ compilers to generate them. It is not required that you use a particular C++ compiler. It is, however, essential for grading that the grader be able to compile and run your programs on their

machines. **It is your responsibility to make sure your program submission can be compiled by the grader. Details on the compiler and settings used by the grader will be provided at the start of the semester.**

If you intend to use your own computers to write the class assignments, it is important that you get a compiler downloaded, installed, and running on your computer as soon as possible. If you don't have a computer, or if you're having problems getting a compiler installed, you should write your programs in the labs until the problems are resolved. In any case, please note that you are responsible for getting the programming assignments written and turned in on time. Since there are many computers available on campus, problems with your personal machines will not be accepted as an excuse for not doing the assignments or late submissions.

Help Desk

For help with issues regarding your computer, UTD maintains a walk-in help desk. Visit their Web site for details:

<http://www.utdallas.edu/ir/helpdesk/>

Schedule (may be adjusted as needed)

- From May 30 to June 20: Review of syllabus, chapters 8, 9, 10, 11, review for test #1.
- Test # 1: June 22, from 3 to 4 PM, at the Testing Center.
- From June 27 to July 20: Chapter 13, 14, 15, review for test #2.
- Test # 2: July 25, from 3 to 4 PM. There may be a class lecture after the test, from 4:30 to 5:15 PM, but that is to be confirmed.
- From July 27 to August 10: elements of C language, chapters 16, 17, 19 and chapter 12 or 18, as time permits, review for test #3.
- Test # 3: August 12, from 3 to 4 PM. Test # 3 is the Final exam.

University's Policies and Procedures

Please go to <http://go.utdallas.edu/syllabus-policies> for information on the university's policies and procedures, which include in particular:

- Student Conduct & Discipline
- Academic Integrity
- Withdrawal from Class
- Student Grievance Procedures
- Incomplete Grade Policy
- Disability Services
- Religious Holy Days

These descriptions and topics are subject to change at the discretion of the Instructor.