	Course	CS 3376.0W1 SE 3376.0W1
	Professor	Dr. Richard Min (Ph.D., MBA, MS, MDiv, STM)
	Term	Fall 2015
	Meetings	Online course except for tests at UTD Testing Center (Student Assessment Center, McDermott Library 1st floor) Test1 10/09 Friday (2 hours from 9am to 8pm) Test2 12/04 Friday (2 hours from 9am to 8pm)

Professor's Contact Information

Office Phone	972-883-4522
Office Location	ECSS 4.609
Email Address	Richard.Min@utdallas.edu
Office Hours	Monday & Wednesday 12:30-2pm, 4-5:30pm, 7:30-9:30pm (or by appointment)

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

100% online course delivered through elearning.utdallas.edu
(Login to access lecture notes, assignments, discussions, grades, etc.)

Course Pre-requisites, Co-requisites, and/or Other Restrictions

Computer Science I and II (CS 1337 and CS 2336 or equivalent courses)
Basic Object Oriented Programming Skills

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. Prerequisite: ECS 2336 or equivalent. (3-0) S

Student Learning Objectives/Outcomes

1. Ability to create classes of abstract data consisting of variables and functions
2. Ability to utilize C++ constructors, copy constructors, and destructors
3. Ability to utilize C++ OOP features using static member data and member functions
4. Ability to utilize C++ File and Stream Input/Output Processes
5. Ability to generate reusable code using inheritance
6. Ability to use polymorphism and virtual member functions
7. Ability to generate reusable code using templates
8. Ability to create and utilize dynamic data structures such as linked lists
9. Ability to create and utilize recursive functions

Required Textbooks and Materials

1. Gaddis, Starting Out with C++ From Control Structures through Objects 8e (7/e is OK, 0132576252)
   (This is the textbook for your cs1336 and cs1337. We will review chapters 9-19).

The following books are available (online and free access) via UTD eBook Safari.

2. Beginning Linux Programming, 4th edition by *Neil Matthew, Richard Stones*
   ISBN-10: 0470147628 ISBN-13: 978-0470147627
   (Also available online & free via UTD eLibrary => Safari)

3. *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: 978-0-13-308504-4
   (Also available online & free via UTD eLibrary => Safari)

4. *Advanced Programming in the UNIX® Environment*, 3e. W. Richard Stevens and Stephen A. Rago.
   Addison-Wesley. © 2013. ISBN-10: 0-321-63773-9.
   (Available also via UTD ebook – Safari).

5. *C++ Programming Language.* 4/e. Stroustrup ©2014 Addison-Wesley ISBN-10: 0321958322.
   ISBN-13: 9780321958327
   (Available also via UTD ebook – Safari).

Suggested Course Materials

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

Academic Calendar: See "Course Homepage" within elearning for the detailed schedule. It will be updated with lecture notes as the semester proceeds.

Grading Policy

[Note: The following items and dates here are tentatively assigned and are subject to change as needed.]

Course credit is given only for the work assigned in the course schedule. There will be no opportunities for any extra credit in this course. The final grade will be computed as follows:

Tests	50%	Test 1: 25% & Test 2: 25% Each test will be taken at Testing Center (Student Assessment Center, McDermott Library 1st floor) on 10/09 Friday and 12/04 Friday (from 9am to 8pm, for 2 hours). 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 on Saturdays. 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.

		All make-up tests will be scheduled during the week following 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 prior to the test date except for serious medical condition (Doctor certificate or hospital receipt will be required as 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 40%. 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.
Assignments (Projects)	20%	4 Assignments (projects) contributing 5% each. Due: (1) 9/24 Th, (2) 10/22 Th, (3) 11/19 Th, (4)12/08 Tuesday 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. 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 projects on time. My 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. All these 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 project submission steps will be given with project #1.
Weekly Post (Discussion)	10%	2 meaningful 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!" or "Weather is bad" etc. will not count towards participation. Your post can be a good question, meaningful response to another student's question, interesting observation, etc. 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.
Weekly Activity (Activities & Quizzes)	20%	Weekly assignment (including online quiz) will be posted by Monday & will be due Sunday midnight every week. It will be small programming exercise in most weeks. It can also be a quiz or some other activity as well. It will vary every week. Each one will be scored out of 2 points. Reasonable progress towards the expected answer will get 1 point & perfect or near-perfect submissions will get 2 points. Late submissions are NOT accepted for weekly assignments.

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				

Weighted total in your gradebook shows the current weighted grade based on your graded work. For example, if you have done only test 1, 2 programming projects, 3 weeks of online participation & 3 weekly assignments so far, current grade will be based on only those entries.

Course & Instructor Policies

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 project, it is better to turnin 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 projects/weekly assignments/tests in a week 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.

Assignments/Projects will be graded on a 100 point basis, utilizing the following criteria:

		Max Score
Source Code	Overall design	40%
	Comments & Indentation	10%
	Demo	10%
	Documentation in Word file (with Code & Screenshots)	10%
Execution	Nominal cases	25%
	Special cases	5%
Total		**100%**

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 tohttp://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.