CS 2308: Foundations of Computer Science II
Spring 2013

Section 255
Section 256

Instructor: Dr. Jill Seaman
Nueces 221
245-4706
js236@txstate.edu

Course Webpage: http://www.cs.txstate.edu/~js236/cs2308

Office Hours: MW: 12:30PM – 1:30PM
TR: 3:00PM – 4:30PM
and by appointment.
Subject to change (if so you will be notified).

Meeting Time/Place: Section 255: TR 5:00PM-6:20PM DERR 240
Section 256: MW 2:00PM-3:20PM DERR 241

Open Labs: DERR 231: Linux Lab
MCS 590: Windows Lab
MCS 594: Lab tutors

Text: Tony Gaddis, Starting out with C++: From Control Structures through Objects, 7th Edition, ISBN: 0132576252

List of recommended/required readings:
Chapters 1-7 (review of CS 1428)   (recommended)
Chapters 8,9,10,11,13,14,17,18   (required)

Prerequisites: C or higher in CS 1428: Foundations of Computer Science I

Course Description: Fundamentals of object-oriented programming. Introduction to abstract data types (ADTs) including lists, stacks, and queues. Searching and sorting. Pointers and dynamic memory allocation. A continuation of CS 1428.

Course Objectives:
1. Develop and use appropriate algorithms.
2. Know that there are typically many algorithms for the same task (for example, searching and sorting).
3. Implement a divide-and-conquer algorithm to solve an appropriate problem (binary search).
4. Have an introductory knowledge of the time/space efficiency of various algorithms.
5. Understand structured programming in terms of modules and functions.

6. Understand how to separate source code into multiple files, including header (.h) files.
7. Use pointer variables and memory operations.
8. Resize an array with dynamic memory allocation.
9. Delete allocated memory to avoid memory leaks.
10. Create and use simple linked-lists.
11. Insert into, delete from, and traverse a linked structure.
12. Understand the principle of the Abstract Data Type (ADT) and, in particular, the separation of interface and implementation.
13. Implement user-defined data structures in a high-level language.
14. Compare and contrast the costs and benefits of dynamic and static data structure implementations.
15. Have an introductory understanding of object-oriented programming.
16. Write a program using an array of objects.
17. Design, implement, test, and debug simple programs in an object-oriented programming language.
18. Describe how the class mechanism supports encapsulation and information hiding.
19. Write programs that use each of the following data structures: arrays, structures, strings, and linked lists.
20. Describe and understand concepts of Stacks and Queues.
21. Be able to create, compile, and run a program in a Unix style, command-line environment.

Grading:

Attendance:		required	
Quizzes:	5%	6-8 total	
Programming Assignments:	25%	7-8 total	
Exam I:	20%	Feb 25 (M) and Feb 26 (T)	
Exam II:	20%	Apr 10 (W) and Apr 11 (Th)	
Final Exam (comprehensive):	30%	See below	

Final Exam Dates: Section 255 (TH):   Thur, May 2, 5:00PM to 7:30PM
                           Section 256 (MW): Mon, May 6, 2:00PM to 4:30PM

Attendance: I record attendance every day and I expect you to be in class every day. However, it is not part of the calculation of your final grade.

Quizzes: Quizzes are usually announced during the previous class and will count for 5 points each.

Makeup Policy: Missed quizzes and Programming assignments cannot be made up. Exams may be made up in exceptional circumstances, with documentation and/or approval from the instructor.

Late policy for programming assignments: see the class webpage.

Notifications from the instructor: Notifications related to this class will be sent to your Texas State e-mail account. Be sure to check it regularly.

TRACS: We will use the TRACS website for the following:
- Grades (Gradebook2 tool)
- Programming assignment submissions (Assignments tool)
- Programming assignment solutions (Resources tool)

Everything else will be on the class webpage (including lecture presentations)

Withdrawals/drops: You must follow the withdrawal and drop policy set up by the University and the College of Science. You are responsible for making sure that the drop process is complete.
http://www.registrar.txstate.edu/registration/drop-a-class.html

Last day to drop: March 21, 2013.

Classroom Behavior: The main rule is to not disrupt or distract other students during class. Please do not arrive late or leave early (without prior permission from the instructor). Cell phones, iPods, etc. should be kept out of sight and turned off or on vibrate during lecture.

Academic Honesty: You are expected to adhere to the University's Academic Honor Code as described in http://www.txstate.edu/effective/upps/upps-07-10-01.html. Also see the Texas State Student Handbook.

All assignments are to be done individually! You may discuss general strategies for attacking assignment problems with other students in the class but **you must write your own code**.

Do not email your program to anyone (except the instructor)!

The penalty for submitting a program that has been derived from a common ancestor of another students' program will be a 0 for that assignment.

Your submitted programs may be run through an internet service designed for detecting plagiarism in software code.

Accommodations for students with disability:
Any student with a special needs requiring special accommodations should inform me during the first two weeks of classes. The student should also contact the office of disability services at the LBJ student center.