CS 308 Data Structures

Spring 2003

Prerequisites: (Old Catalog) CS202 and CS236 (New Catalog) B or better in CS201 and CS202. If you do not meet the prerequisite requirements for this course, you should see me immediately. *Credit hours:* 3.0

Instructor: Dr. George Bebis
Office: 235 SEM
Phone: 784-6463
E-mail: bebis@cs.unr.edu
WEB: http://www.cs.unr.edu/˜bebis
Office Hours: TR: noon - 1:00 PM and by appointment

Required Text:

C++ Plus Data Structures by N. Dale, Jones and Bartlett Publishers, 2nd edition, 2001.

Optional Texts:

Practical Debugging in C++, by A. Ford and T. Teorey, Prentice Hall 2002.
Data Structures with C++, by W. Ford and W. Topp, Prentice Hall, 1996.
Data Structures and Program Design in C++, by R. Kruse and A. Ryba, Prentice Hall, 1999.
Data Structures and Other Objects Using C++, by M. Main and W. Savitch, Addison-Wesley, 1997.

Objectives

The purpose of this course is to introduce you to the exciting world of data structures, an issue central to the art of computer programming. At the end of the course you will be equipped with the tools of data organization to enable you to write simple, clear, and efficient programs. The course will be structured around a comprehensive set of computer assignments to enable you to get hands on experience. Our programming language of choice will be C++. Hence, it is required that you have some programming experience in C++.

Course Outline (tentative)

- Software Engineering Principles (read Chapter 1)
- C++ Review
- Unsorted and Sorted Lists
- Stack and Queue
- Linked Structures
- Lists Plus
- Recursion
- Binary Search Tree
- Trees Plus
- Sorting and Searching (if time permits)

Exams and Assignments

There will be one midterm exam and a final. The material covered by the exams will be drawn from the lectures and from the homework. Both exams will be closed books, closed notes. Moreover, there will be 5-6 quizzes in class (they will be announced at least one class period in advance). Homework problems will be assigned but will not be collected for grading. Also, there will be 4-5 programming assignments. The goal of the assignments is to provide you with hands-on experience with handling various kind of data structures. In the lectures, you will be introduced to data structures at an *abstract* level and in the assignments you will write code to *implement* and *use* these data structures. The set of tasks chosen for your assignments all relate to *images* (check the web page of the course for more information on how to read, write, and display images). You will design algorithms to perform simple image processing tasks using data structures such as arrays, stacks, queues, trees, and hash tables. These exercises are not only important from a grading standpoint but also are crucial learning tools.

Course Policies

Lecture notes, programming assignments, and other useful information will be posted on the course web page (*http://www.cs.unr.edu/˜bebis/CS308*). You should check the web page regularly. The lecture notes will be in powerpoint.

You will use your *Linux* accounts to do the assignments. The lab contains a number of PCs to host your assignments and is a good place to do your work. You can, of course, remote log in to any of the PCs. The assignments are to be completed in groups of **2-3**, however, all the members of the team are expected to fully understand the structure of the code and the implemented algorithms. Discussion of the programming assignments is allowed and encouraged. However, each team is expected to do its own work. **Assignments which are too similar will receive a zero.**

Regular attendance is highly recommended. If you miss a class, you are responsible for all material covered or assigned in class. **Late programming assignments will be penalized 10% of the points assigned per day (weekends count as one day).** If you are unable to hand in an assignment by the deadline, you must discuss it with me **before** the deadline in order to avoid the late penalty. **No incomplete grades (INC) will be given in this course and a missed exam or quiz may be made up only if it was missed due to an extreme emergency.**

Useful Tips

Since the material in this course is highly integrated, a limited understanding of one topic will have a serious effect on the understanding of subsequent topics. You should expect to spend many hours on this course outside the classroom. Do not expect to fully understand the material covered in this class if you do not spend many hours in front of your computer.

Don’t get behind in the programming assignments. Probably the main reason for students doing poorly in this course is getting behind in the assignments and never recovering. Design and implement in a top-down, modular fashion. Get something working that has the skeleton structure of what you need and then add features to it. Each time you add a feature, test it and make sure everything is still working. It can be tough to debug big programs if all you know is that the output is wrong and you are not sure any one module is working. In addition, partial credit will be given for a program which at least partially works while it is very difficult to give credit for a program which may have many features but is not doing anything correctly.

Grading Scheme

Midterm: 20%
Final : 20%
Quizzes: 20%
Programming Assignments: 40%

A 90 and above
B 80-89
C 70-79
D 60-69
F < 59

Important dates

3/11/2003 - Midterm
3/14/2003 - last day for dropping classes
3/15/2003 to 3/23/2003 - Spring Break
5/8/2003 - Final exam (2:15pm - 4:15pm)