

CS 302 Data Structures

Spring 2012 – Dr. George Bebis

Catalog Description: Data structures and algorithms fundamental to computer science; abstract data-type concepts; measures of program running time and time complexity; algorithm analysis and design techniques.

Prerequisites: C or above in CS202. If you do not meet the prerequisite requirements for this course, you should see me immediately. *Credit hours:* 3.0

Meets: TR 2:30 PM - 3:45 PM (AB 209)

Course Webpage: <http://www.cse.unr.edu/~bebis/CS302>

Instructor: Dr. George Bebis

Office: 235 SEM

Phone: 784-6463

E-mail: bebis@cse.unr.edu

Office Hours: Tuesday/Thursday 4:00 pm - 5:30 pm and by appointment

Student Assistant: Emily Hand

E-mail: e.hand@live.com

Office: ECC (Linux Lab), SEM 2nd floor

Office Hours: Wednesday 4:00pm – 6:00pm and Friday, 1:00pm – 4:00pm.

Required Text:

C++ Plus Data Structures by N. Dale, Jones and Bartlett Publishers, 4th edition, 2007.

Optional Texts:

Data Structures with C++ by W.Ford and W.Topp, Prentice Hall.

Data Structures and Program Design in C++ by R.Kruse and A. Ryba, Prentice Hall.

Data Structures and Other Objects Using C++ by M.Main and W.Savitch, Addison-Wesley.

Objectives

The purpose of this course is to introduce you to 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++.

Course Outline (tentative)

- C++ Review
- Introduction to Image Processing and Analysis
- Analysis of Algorithms
- Stacks (i.e., array-based an linked-list-based) & Templates
- Queues (i.e., array-based an linked-list-based)
- Unsorted Lists (i.e., array-based an linked-list-based)
- Sorted Lists (i.e., array-based an linked-list-based)
- Recursion
- Binary Search Trees
- Heaps and Priority Queues
- Graphs (i.e., array-based an linked-list-based)
 - Searching (i.e., DFS and BFS)
 - Shortest Paths (i.e., Dijkstra's and Bellman-Ford algorithms)
- Sorting
 - Selection Sort, Bubble Sort, Insertion Sort
 - Heapsort, Mergesort, Quicksort
 - Linear Time Sorts
- Advanced Data Structures for Searching
 - Range Trees
 - kD-Trees
 - Quadtree
 - Hashing

Exams and Assignments

Grading will be based on 2 exams, 6-7 quizzes, and 4-5 programming assignments. Homework will be assigned but will not be collected for grading. Solutions will be provided to all homework problems. Details are provided below:

- Quizzes will be announced at least one class period in advance.
- There will be a midterm and a final exam. The material covered in the exams will be drawn from the lectures, quizzes, and homework.
- Programming assignments will be done in groups of two students. The goal is to provide you with hands-on experience with handling various kinds 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.

Course Policies

- Lecture slides, assignments, and other useful information will be posted on the course web page.
- Regular attendance is highly recommended. If you miss a class, you are responsible for all material covered or assigned in class.
- A missed quiz or exam may be made up **only** if it was missed due to an extreme emergency.
- The programming assignments will be completed in groups of **two**. Both team members are expected to fully understand the structure of the code and the implemented algorithms. Discussion of the programming assignments is allowed and encouraged. However, team members are expected to do their own work. **Assignments which are too similar will receive a zero.**
- **No** late programming assignments will be accepted unless there is an extreme emergency.
- **No** incomplete grades (INC) will be given in this course

Extra Credit

Class participation is highly encouraged and will be rewarded with extra credit. Also, extra credit will be offered to the students who attend the departmental colloquia. You will be reminded in class about upcoming talks but you should also check the colloquia page on a regular basis (<http://www.cse.unr.edu/get-involved/colloquia/>)

Useful Tips

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 anyone 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.

Academic Dishonesty

Your continued enrollment in this course implies that you have read the section on Academic Dishonesty found in the UNR Student Handbook (see <http://www.unr.edu/stsv/acdispol.html>) and that you subscribe to the principles

stated therein.

Disability Statement

Any student with a disability needing academic accommodations is requested to speak with me or contact the Disability Resource Center (Thompson Building, Suite 101), as soon as possible to arrange for appropriate accommodations.

Unauthorized class audio recording or video-taping

Surreptitious or covert video-taping of class or unauthorized audio recording of class is prohibited by law and by Board of Regents policy. This class may be videotaped or audio recorded only with the written permission of the instructor. In order to accommodate students with disabilities, some students may have been given permission to record class lectures and discussions. Therefore, students should understand that their comments during class may be recorded.

Grading Scheme

Midterm: 25%
Final: 25%
Quizzes: 20%
Prog. Assignments :
30%

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

Important dates

March 15, 2012 – Midterm exam
March 23, 2012 – Final Day to Drop Classes
March 17-25, 2012 – Spring Break (no classes)
May 9, 2012 – Prep Day
May 10, 2012 - Final exam (12:30pm – 2:30pm)