CS361/CS561N

Data Structures

Credit:  5
Instructor:  Dr. Robert P. Judd
Email:  juddrb@bobcat.ent.ohiou.edu
Home page:  http://oucsace.cs.ohiou.edu/~juddrp/cs361/
Office:  349 Stocker
Office hours:  2-3 MTThF
Prerequisite:  CS240C, CS300

Objectives:  To become proficient designing and analyzing complex data structures and algorithms. In particular to understand and design programs utilizing fundamental data structures such as lists, trees, graphs, and heaps, and be able to analyze both the space and time complexities of such algorithms. To understand and design programs utilizing searching and sorting algorithms based on these data structures.


Grades:  Your grade will be determined by a combination of your homework and programming projects (30%), two one hour exams (35%), and the final (35%). You must pass both the exam and project portions of the course to pass.

Academic Conduct:  All work must be your own. You cannot exchange written, electronic or oral representations of either your code or pseudo code for any of the programming projects. Any violation of this policy will be pursued and can result in a penalty ranging from failure of the assignment or exam to failure of the course.

Previous Background:  It will be assumed that you know

- how to program in C++;
- concepts of object-oriented programming;
- templates;
- stacks, queues, and lists;
- complexity analysis (big O);
- how to use STL, and
- the UNIX and prime environment.
Course Outline: The course emphasizes the importance of fundamental data structures and algorithms to programming. Here we build on CS240C, and examine the following topics in more detail:

- abstract data types
- mathematical analysis of algorithms
- trees and graphs
- searching and sorting techniques
- advanced data structures

Program grading: In general, your program will be graded based on how well it processes valid data and how efficient it is.

- It must compile on the Sun Solaris OS using g++.
- Many times your assignments will be to create classes with predefined method signatures. I will then test these methods against my own main. The signatures cannot be changed. For these assignments, I will distribute a sample main program. If your submitted classes do not compile against this main, no credit will be awarded.
- Your program will be tested against several data sets. A point value will be assigned to each set. A maximum of 30% of this point value will be awarded if your program terminates with a wrong answer, with a runtime error, or enters an infinite loop.
- Up to 50% of a program grade may be deducted based on inefficient programming methods, poor style, or violation of programming rules.