CS 361 - Introduction to Data Structures

The Old Dominion Honor Code is very important. Please obey all rules and regulations. Please do not be afraid to assist and learn from each other. But you must have personally written ALL code submitted for evaluation. Sophisticated software will be used to check for "borrowed" code. All students are expected to use standard security protocols on their university directories.

Perhaps more than any other course, CS361 is intended to expand the students’ toolbox of basic techniques for manipulating data, the student experience a variety of standard practices and approaches used in program design. At the concrete level, the student will begin what should be a career-long practice of accumulating useful, reusable code units.

This course explores data structures, algorithms for manipulating them, and the practical problems of implementing those structures in real programming languages and environments. Heavy emphasis is placed upon the analysis of algorithms to characterize their worst and average case requirements for running time and memory.

Late work will be assessed a 15% penalty per day each day for 3 days after that the work will receive a mark of 0.

Grading Criteria

Assignments and Quizzes: 60%
Midterm Examination: 20%
Final Examination: 20%

Prerequisites will be strictly enforced

CS 250, Problem Solving and Programming,
CS 252, Introduction to Unix for Programmers
MATH 163, Pre-Calculus II
For your convenience the link to CS252 is included here: CS 252

Detailed Topics List

Brief review of classes
Design of objects  (Professor Zeil's Design Checklist)
The Link List revisted
Implementing the Linked List ADT using the STL
Algorithms - best, worst, and average case big O analysis.
Overview of various data structures
  - queues
  - priority queues
  - deques
  - vectors
  - stacks
  - heaps
  - sets and multiset
  - trees, binary and n-ary
  - graphs
  - maps and multmaps