CS 2420 Algorithms and Data Structures

Instructor: Vicki Allan

Office: Main 429

Communications: Phone: 797-2022, E-mail: Vicki.Allan@usu.edu. This course is being video-taped for distance students. Class lectures are available at tcs.uen.net/tcs. I find I need to download the lecture before playing it. Late afternoon office hours are available to accommodate distance students. Email will normally be answered within twelve hours, excluding weekends.

Office Hours: MWF 11-12 a.m., 4-5 p.m. Additional office hours are available by appointment or when my office door is open.

Prerequisite: CS 1410 or equivalent knowledge.
If you feel that you are a poor C++ programmer, your chances of succeeding in this class are slim. I would strongly recommend that you take CS1410 instead of this class. I expect that you know the following material: structs, unions, file operations, classes, friends, overloading operators, inheritance, polymorphism, virtual, exceptions, templates, linked lists, stacks, queues, recursion, and binary trees. There is so much new material to be learned, you do not have time to learn material that you should have mastered in earlier classes. Sometimes students like to gamble on success. The laws of gambling state that the smaller the probability of succeeding, the greater the reward. Many times the gamble in taking a class (given the time you are willing to spend and the skills you bring to it) has a very low probability of success and a small reward (you may barely pass, but the hours required may cause you to do poorly in other classes).

Every semester I see many students throwing away time and tuition money by taking classes they can't possibly pass with their skills and/or their commitment. They go through the motions of taking the course, but don't learn what they need to. Some students say, "The instructor is so nice. If I come every day and try my best, surely she/he will give me a passing grade." It won't happen. Giving you a passing grade when you haven't earned it is not only unfair to others, but it is no favor to you. You would go on to struggle in other classes.

Tutors: Tutors for data structures are available in Main 425. Please use them whenever possible. If a tutor is on duty, I expect that you will consult with her/him before you come to see me. The hours for fall semester are:

Monday through Friday: 11:30-9:00pm
Saturday: 12:00-5:00pm

If there are tutors that are especially good or bad, please let me know about it. We want the quality of the tutors to be high.

Course Page: All assignments and other information dealing with the course are posted on the course page www.cs.usu.edu/~allanv/cs2420/cs2420.htm

Text: *Data Structures and Algorithm Analysis in C++, 3rd Edition* by Mark Weiss. Addison-Wesley, 2006(ISBN 0-321-44146-X).

Objectives: The main objectives of this course are: a mastery of data structures, a mastery of refining programming skills, a mastery of developing strategies for the design and evaluation of algorithms, a familiarity with algorithm analysis, a mastery of recursion, a familiarity of sorting algorithms, a familiarity of graphs, a familiarity of trees, a mastery of binary search trees, a mastery of hash tables, a mastery of priority queues, a mastery of splay trees, a mastery of merging priority queues, a familiarity of disjoint set classes, an exposure to dynamic programming, an exposure to greedy algorithms. Memorization or copying is not learning and will not be encouraged. Class discussion utilizes discovery learning and will be very different in nature from the step by step, cookbook approach of most texts. Since you will experience both the text's presentation and the derivation of the ideas in class, you will have the benefit of both teaching techniques.

Requirements:
Programming Assignments: There are seven programming assignments during the semester. The point values of the programming assignments are not commensurate with the time involved to complete them. Programming assignments should be viewed as essential preparation for exams, rather than work that is adequately rewarded. All programs are to be written using C++. You may use any computer system you desire. Each program must be your own work (this includes **not** allowing a tutor to write your programs). You are **not** allowed to use the STL (standard template library) unless the assignment explicitly states you can.

All programming assignments are to be turned in using the eagle system. Go to the eagle system and enroll in the class. All communication about the class and/or grading should be directed to me (Vicki.Allan@usu.edu). Assignments turned in after 11:59:59 p.m. on the date due are late. Students are responsible for turning in their programs on time.

I rarely alter the due date of an assignment, and will not do so unless all students can be informed of the change at least two days before the original due date. Computer failures and file loses are a part of dealing with computers and will not be considered an excuse except in extreme circumstances.

Use the Style Guidelines available from the class web page. Your programs are graded based on these guidelines; make sure you understand them. You will lose points for violating the standard.
Programming assignments are graded as follows:

40% Program contains no functional errors and produces correct output.
30% Efficient, well designed, extendible code.
15% Readability, good variable names, readability.
10% Comments. Well commented source code is often a necessity for others who will read your code. This includes explanation of variable names, functions, and descriptions of chunks of code. Note, comments should not be on every line.
5% Format of output is pleasing and easy to understand. A person should be able to tell what the program does my just looking at the output. Put enough information in the output so this is true.

Programming is an important part of this class. You cannot receive higher than a D+ if you are missing any programming assignments or you have less than 60% of the programming points. This is true regardless of the total points earned.

Written Assignments: We will have eight written homework. Written homework takes the following form. Specific questions are posted on the course web page. You may work in groups of one, two, or

three. Groups may change throughout the semester. Feel free to visit with me about possible answers to homework exercises. If more than one person is involved, list all the names on ONE set of answers. Answers should not be compared with others not in your group. Thus, it is cheating if you visit with others about answers, but turn in the homework with only your name on it. The written assignment must be in one of the following formats: .doc, .docx, .pdf, or .odt.

You will learn much more by working in a group than you will learn working by yourself. Educationally, it is a superior experience. You have to defend your answers. You get to take turns explaining and being taught. There are more of you to seek help from me, should you need it. When you do seek help, you are more confident that you have an important question as there are three of you with the same question. Thus, you don't feel "It's just me." Instead of just skipping a question you don't understand, you are able to iterate through several choices. You come to class having really worked on every question. For good students, it will increase your understanding. For poor students, it may be the only way that you survive. I realize there are some of you who absolutely have no time to work with others, but these situations should be rare.

Exams: There are two midterm exams (100 points each) and the final (150 points), given on October 11th, November 12th, and the final, which is comprehensive, given on Wednesday, December 15th, at 11:30am. Exams cover material presented in class, in the book, and on the programming assignments. I do not give makeup exams. The final is comprehensive, is not optional, and is not given early. Please verify that you are able to take all the exams on the date specified.

Grading:

Programming Assignments	140
Written Assignments	80
Exams	350
Attendance and Preparation	25
Total	**595**

You may be given an F if your overall average is below 50%. Remember, you cannot receive higher than a D+ if you haven't turned in all the programming assignments or don't have over 60% of the possible points on the programming assignments.
Current scores can be checked on the eagle system. Special announcements about the class are e-mailed to you, so it is suggested that you check your e-mail daily.

Course Outline:

Chapter	Topic
2	Algorithm Analysis - sections 2.1, 2.2, 2.3, 2.4
4	Trees - sections 4.1, 4.2, 4,3, 4.4, 4.5, 4.6, 4.7
5	Hashing - sections 5.1, 5.2, 5.3, 5.4, 5.5
6	Priority Queues (Heaps) - sections 6.1, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7, 6,8
7	Sorting - sections 7.1, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7
8	The Disjoint Set Class - sections 8.1, 8.2, 8.3, 8.4, 8.5, 8.6, 8.7
9	Graph Algorithms - sections 9.1, 9.2, 9.3, 9.4, 9.5, 9.6