Reminders and announcements:

- The first hour of class on Wednesday, March 21, and Thursday, March 22, has been set aside as a review for Exam 2. Come prepared to ask and answer questions about the exam material.

- Programming Project 1, Part A and Programming Assignment 6 are due at regular time on Wednesday, March 21, and Thursday, March 22.

- Due to the size of the classes, half of the class will be taking the exam in KC-255 so that everyone will have adequate desk space. Your instructor will tell you who is to be in which room during the exam review.

Exam 2 will be on Monday, March 26, and Tuesday, March 27. You may bring one 8.5in x 11in sheet of paper with notes on (only) one side to the exam. You may print out the sheet, but it must be in a 9-point font or larger. E.g., please do not photoreduce or print 4 pages on 1 side. If you handwrite your notes, they can be as small as you like. You may handwrite notes in the margins of a printout.

The exam will be cumulative and comprehensive in the sense that you are expected to be able to read and write code or analyses and designs using concepts from previous chapters. However, the exam will emphasize the material in Chapters 7 and 10; material covered in lectures and in-class exercises through March 19/20; material in homework 4; programming assignments 5-6, and programming project 1. Chapters 6 (file I/O and formatting) and 8(strings) will not be on the exam.)

The format of the exam will be similar to that of in-class exercise questions, homework questions, and programming assignments. That is, you will be asked to write analyses and designs for programs and/or functions to solve particular problems, and you will be asked to explain and write C++ code examples.

The following is a list of topics that will be emphasized, but it is in no way to be construed as an exclusive list.

1. Analysis, design, and implementation of main programs and functions using the following constructs.
2. One dimensional and multi-dimensional arrays, including declaring arrays, accessing array elements, arrays and array elements as function parameters, functions that process array data.
3. Searching one-dimensional arrays; sorting one-dimensional arrays using the Selection Sort algorithm.
4. Classes - design and implementation of simple classes and use of class objects.

When you are asked to write code, you will not need to write comments, function prototypes, include directives, or output formatting beyond producing newlines in appropriate places.