Reminders and announcements:

- The first hour of class on Thursday, February 8, and Monday, February 12, has been set aside as a review for Exam 1. We will go over Homework 3 as part of the review. Thus **Homework 3 is due by the beginning of class.** **NO LATE SUBMISSIONS** will be allowed for this assignment. In addition, come prepared to ask and answer questions about the exam material.

- Programming Assignment 4 is due on February 13/14 (same day as the exam), by 4:30pm (i.e. regular time).

- **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 1 will be on Tuesday, February 13 and Wednesday, February 14. 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 consist of questions on the material in Chapters 1-5 (excluding Sections 4.6 and 5.4-5.5); material covered in lectures and in-class exercises through February 6/7; material in homeworks 1-3; and programming assignments 1-4. Material on function overloading, output formatting, and debugging 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 and design of main programs and functions. Differences between received, returned, passed back, and local function objects.
2. Declaration and use of constants and variables, use of arithmetic, relational, and logical operators, assignment statements.
3. Basic use of input and output statements (i.e., format manipulators will not be on the exam).
4. Use of selection constructs: single and multi-branch if constructs, case constructs and their implementations.

5. Use of repetition constructs: pre-test loops, post-test loops, counter-controlled loops and their implementations.

6. Implementation of functions, including how to define and use them, (actual) arguments vs. (formal) parameters, value vs. reference parameters, and the use of the **void** type.

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.