CSCI 121 – Spring 2006
Computer Science II

Ben Coleman
coleman@cs.moravian.edu
214 PPHAC
AIM: bjcoleman15

Office Hours: MWRF 10:00 – 11:00
or by appointment
Office Phone: 610-625-7781
Please do not call me at home

Course Description

A continuation of Computer Science I with emphasis on data and procedural abstraction. A study of basic organizations of instructions and data as realized in both hardware design and software development. Topics include encoding schemes for instructions and data, representative machine architectures, data representations in computer memory and in high-level languages.

Goals

- Write programs in C++ using objects, templates, operator overloading, and other advanced language features
- Design programs using object-oriented design, the Standard Template Library, and recursion
- Analyze the performance of algorithms using asymptotic notation, recurrences, and mathematical sums
- Represent data in binary, two's complement, floating point notation, and other representations
- Use tools such as `gdb`, `gnuplot`, and LaTeX

Required Text

- *Problem Solving with C++: The Object of Programming* fifth edition, by Walter Savitch

Responsibilities

Your attendance is expected at each class meeting. You are also responsible for the contents of reading assignments, handouts, lectures, and class email.

If you have a disability that may affect your participation in this course, please contact me immediately to discuss academic accommodations.

Graded Materials

Homework

Approximately every week, you will receive a set of problems that asks you to apply the current material. You are *strongly* encouraged to work together on these problems, but you must write up your own responses.

These solutions should be written in the "other student context." You should provide enough written explanation so that another student in the class, who didn't know how to complete the exercise, could read the submitted material and, without asking questions, understand what should have been done. Written responses which consist entirely of printed source code (even if bullet proof and well documented) will be considered incomplete and poorly written.

Projects

Programming assignments will be assigned throughout the semester. These projects emphasize all aspects of program development, including design, implementation, testing, and documentation.

You will have at least a week to work on each project, but it is especially important that you begin these assignments as soon as they are announced. Late submissions will be penalized ten percent per day. This deduction *may* be avoided by discussing your progress with me *before* the deadline.

You cannot pass the course unless you turn in all projects.

Labs

Each Thursday class session will be used for a lab session. Labs will introduce a variety of tools, and explore the current material from the classroom sessions. Lab work will be due each Monday at 11:59 p.m. As with projects, late work will be penalized ten percent per day.

Tests

Three tests will be given during the semester. The *tentative* dates for these tests are Monday, February 6, Wednesday, March 1, and Friday, March 31. Each test will be given in class, and may only be re-scheduled for medical or family emergencies. In such a situation, you must see me *before* the time of the test.

Grading

Homework	15%
Projects	25%
Labs	20%
Tests	25%
Final	15%

All grades will be calculated on the standard scale using pluses and minuses.