

Professor: Torgersen **E-mail:** *torgerse@wfu.edu* **Office Phone:** 758-5536

Office/Hours: Manchester 226, MW 3:00 to 4:30, Th 3:30 to 5:00

Text: Absolute C++ by Walter Savitch.

Facilities: gottlieb.cs.wfu.edu, telesto.cs.wfu.edu, X2Go software

Goals:

1. Problem solving skills
2. Proficiency in using the UNIX operating system
3. Programming skills in C/C++
 - (a) The Art and Feng Shui of Compiler Error Messages
 - (b) Finding and Correcting Program Errors
 - (c) Procedural programming
 - (d) Object-oriented programming
 - (e) Event driven programming (if time allows)

Tools and Topics:

1. Overview of the UNIX operating system.
 - (a) Online tutorial and quiz: <http://menehune.opt.wfu.edu/Unix/index.html>
2. Programming tools: separate compilation, header files, make, IDE
3. Debugging tools: gdb, ddd

C++ Topics:

1. Transition from Java: C++ program structure
2. Transition from Java: variables, declarations, scope, data types, control structures and functions
3. Pass by value and pass by reference
4. Pointers
5. Arrays: fixed-size and dynamically allocated arrays
6. Dynamic memory allocation and error checking
7. Function overloading
8. C-style strings, command line arguments, and arrays of pointers
9. Structures (struct) and Unions
10. Classes: member data, public, private data
11. Classes: member functions, constructors and destructors
12. Classes: friend functions and friend classes
13. Abstract data types and classes; design, encapsulation
14. C++ string class
15. Function and operator overloading
16. Linked lists, doubly linked lists, implementation, and iterators
17. Class templates, function templates (if time)

Topics for Program Development:

1. Good program design
2. Designing Classes: customer/service model
3. Tracing and logic checking
4. Unit testing, integration testing

Problem Solving and Frequently Used Algorithms:

1. Sorting, searching, ordered and unordered lists
2. Binary search
3. Linear data structures: Linked lists, search, insert and delete

Course Organization

Expectations:

1. Class participation.
2. Communicate if things get complicated.
3. Your best effort; keep pace with the course.

Grading:

Four exams (65%), 10 labs, some take home problem sets (35%). Expected exam dates are Wednesday February 9, Wednesday, March 2, Wednesday, April 6, and Tuesday, May 2 (final exam). Programming assignment(s) **must** be submitted ready to compile and run under Linux (Ubuntu). Program assignments typically will require independent work outside of class meeting times.

Disability Notice:

If you have a disability that may require an accommodation for taking this course, then please contact the Learning Assistance Center (758-5929) within the first two weeks of the semester.

Pledge Work Notice:

Assignments in Computer Science courses may be specified as “pledged work” assignments by the professor of the course. When an assignment is specified as “pledged work” the only aid that the student may seek is from either the course professor or an assistant that the professor has explicitly specified. On “pledged work” assignments the student may not use the services of a tutor.

Teamwork:

Outside of a University setting, the software projects you encounter through your career will almost always be developed in a collaborative setting. To facilitate collaborative experience, some assignments may be designated as “teamwork projects”. In these assignments, you may work in pairs. Each student working in pairs is required to write a short assessment of the collaborative experience.