Course Description: This course explores fundamental ideas in the software design and development such as data abstraction, encapsulation, subtyping, and polymorphism. In particular, this course focuses on the expression of those ideas in the C++11 Programming Language, and their application to common programming problems. Design patterns and aspects of software design (modules, libraries, and frameworks) are also covered.

Prerequisites: Completion of 210 with a grade of C- or better.

Course Credits: 3

Books:


Course Objectives: At the completion of this course, students will:

- know the difference between values, objects, and types,
- appreciate the role of types (and classes) to define data abstractions,
- understand and be able to explain “subtyping”,
- understand and be able to explain “polymorphism”,
- know how to encapsulate related ideas into a single class, hierarchy of classes, or module,
- understand interfaces and contracts,
- recognize when to use and how to apply design patterns to certain recurring problems, and
- understand fundamental differences between object-oriented programming languages and their trade-offs.

Grading Scale:
A (93), A- (90), B+ (87), B (83), B- (80), C+ (77), C (73), C- (70), D+ (67), D (63), D- (60), F

Grade Distribution:
Participation 10%, Projects & Homework 50%, Midterm 20%, Final 20%
Attendance: It is in your best interest to attend class.

Homework: Homework will involve reading papers, articles, or blogs and writing a report. Reports will be evaluated on their content, but you will be expected to write professionally (i.e., proper use of language, grammar, and spelling).

Projects will include both C++ implementations and appropriate documentation. Source code will be evaluated on correctness, style, readability, readability, and the application of concepts presented in the course.

Some projects may include collaborative work with other students in the course.

You may also be asked to present and/or demonstrate a final project in front of the class.

Graduate students will be required to complete an additional project, ideally related to their research topic.

Midterm: The date of the midterm will be announced in advance.

Final: The final will be comprehensive.

Policies: The class will be taught using a variety of sources. A primary source will be examples given by the instructor and discussed during class. Notes will be from web sources, the instructors own web pages, and written on the board. Attendance is necessary for complete understanding of the material, and therefore expected.

All work created for this course will be committed as they are developed into a source code repository (i.e., Subversion). The work must appear in the repository in order to be graded. The use of a repository will assist in the communication between the student and the Instructor and in any collaborative projects.

All submitted work must be your own. Submission of work that is entirely or partly not yours will be reported to the Office of Student Conduct.

In order that work can be graded and returned promptly late assignments will not be accepted without a valid excuse. It is up to the student to make up any missed material. Make-ups of any work for this class will only be given in the case of a university-approved excuse. I may be contacted in advance to arrange for known absences.

Notices: Any student who feels that she or he may need an accommodation based on the impact of a disability should contact the Office of Accessibility at 330-972-7928. The office is located in Simmons Hall, 105.

Students whose names do not appear on the University’s official 15-day class list will not be permitted to participate (attend class, take exams, or receive credit).

The University withdrawal policy can be found here: http://www.uakron.edu/ssc/withdrawal-policy.dot