Term: Spring 2016  
Building: Newton Oakes Center room 210, TR 2-3:15pm  
Instructed by: Dr. Tamirat Abegaz: 104 Barnes  
Email: tamirat.abegaz@ung.edu  
Website: http://faculty.ung.edu/tabegaz/  
Office Hours: Tuesday: 11:00AM-12:30pm, Wednesday 1:00pm-4:45pm or by appointment. Subject to change, but you will be notified in advance in the event of any change.

Course Description and Objectives:

The primary objective of this course is to convey the concepts of object-oriented programming (OOP) as they apply to the real world problems using C++. OOP is a conceptual approach to designing and implementing software, and C++ has enhanced C with features that ease the way to realize OOP. The following are the most important OOP features: Abstraction, Encapsulation and data hiding, Polymorphism, Inheritance, and Reusability of code.

This course will cultivate critical thinking skills by presenting object-oriented approach as a means of problem solving. It is intended to sensitize students to the design and implementation of professional software programs fundamentals of computer science, algorithm development, coding, debugging and documentation of programs within the Microsoft Visual Studio .NET environment. The course will also include hands-on activities through assigned programming work. To successfully complete the course, students should allocate up to 10 hours of work outside the class.

Prerequisite: Grade of C or above in CSCI 1302.

Course Outcomes

- To develop an in depth knowledge of computer programming in C++
- To acquire the skill to develop professional software development by applying object oriented principles.
- To understand the Standard Template Library (STL)

Required Materials:


Software


Any and all other required materials, or means by which to obtain these materials, will be physically or electronically provided to you by the instructor.

Topics:

1. Introduction
   a. Understanding the cin and cout objects
   b. Understanding abstraction

2. Understanding classes
   a. Creating classes
   b. Understanding constructors and destructors
   c. Usage of smart pointers
   d. Encapsulating class components
   e. Implementing member functions
   f. Classifying the roles of member functions
   g. Parameter passing (by value, by reference)
   h. Understanding the this pointer

3. Class features and design issues
   a. Understanding Polymorphism
   b. Understanding composition

4. Understanding friends and overloading operators
   a. Creating friend classes
   b. Implementing friend functions
   c. Using operator overloading

5. Understanding Inheritance
   a. Single Inheritance
   b. Multiple Inheritance

6. Handling Exception and File Manipulation
   a. Throwing Exception
   b. Using try blocks
   c. Catching exceptions
   d. Writing objects to the file

7. Using Templates
   a. Creating class templates
   b. Creating function templates
   c. Overloading function templates
   d. Understanding the Standard Template Library (STL)
Methods of Instruction:

A combination of Lecture, Labs, Multimedia, and group discussions are the major methods of instruction that are used in this course.

Important Dates:

January 11 – First day of class
January 11-15 add/drop period
March 10 -- Mid-term Exam
March 11-19 – Spring Break – NO CLASSES
April 29 – Last day of class
May 2 -6 – Final Exams week

Evaluation Methods:

<table>
<thead>
<tr>
<th>Item</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quizzes/ Exercises</td>
<td>10%</td>
</tr>
<tr>
<td>Mid-term Exam</td>
<td>20%</td>
</tr>
<tr>
<td>Assignments &amp; Projects</td>
<td>40%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>30%</td>
</tr>
<tr>
<td>Total</td>
<td>100%</td>
</tr>
</tbody>
</table>

When grades are handed or emailed to you for assignments and projects, you have 7 days to schedule and meet with me for grade changes. After that point, the grades are written in stone and can't be changed.

Grading Scale

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>90% and above</td>
</tr>
<tr>
<td>B</td>
<td>80% and above</td>
</tr>
<tr>
<td>C</td>
<td>65% and above</td>
</tr>
<tr>
<td>D</td>
<td>50% and above</td>
</tr>
<tr>
<td>F</td>
<td>Below 50%</td>
</tr>
</tbody>
</table>

Quizzes may or may not be announced.

The Final Exam covers all the topics and must be taken at its scheduled time. If for any emergency, you need to take the finals before the scheduled date, you must submit your request at least 5 days before the scheduled date.

Lab Requirements and Policies:

Labs are designed for hands-on experience. They are scheduled during the regular class hours and will be treated as regular class hours as far as the attendance policy is concerned.
During the laboratory sessions, you must work on the assigned laboratory programs. If you finish early, you may choose to help others or to leave class early. See the instructor before leaving to avoid missing end of class material.

**Project grading:**

Projects will be graded as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithm (correct and efficient)</td>
<td>25%</td>
</tr>
<tr>
<td>Correct compilation and execution</td>
<td>50%</td>
</tr>
<tr>
<td>Documentation (comments)</td>
<td>25%</td>
</tr>
</tbody>
</table>

**Project Submission Requirements:**

- Your project must compile correctly and produce the desired output.
- You will submit projects online. Submission instructions will be given in class during project assignment.
- As with industry software development, you have to adhere to coding standards followed by the authors in the textbook. Additional requirements will be discussed in class and will be notified as needed.

**Extra Credit Policy**

Extra credit may be provided to students who submitted outstanding work by adding useful features to the project. It should exceed the requirements of the assignment significantly. If you strive for it, three or more extra credit will exempt you from taking the final exam.

**Assignment extensions**

You have three pre-approved late days. You can use it for one, two, or three assignments. Do not use unless it is necessary. No additional extensions allowed (you will get 0) unless you have serious matter.

**Supplemental Syllabus**

Students are required to refer to the Supplemental Syllabus (http://ung.edu/academic-affairs/policies-and-guidelines/supplemental-syllabus.php) for the following important categories information:

- ACADEMIC SUCCESS PLAN PROGRAM
- STUDENTS WITH DISABILITIES
- ACADEMIC INTEGRITY POLICY
- DISRUPTIVE BEHAVIOR POLICY
- CLASS EVALUATIONS
- ACADEMIC EXCHANGE
- INCLEMENT WEATHER
- COURSE GRADES AND WITHDRAWAL PROCESS