Syllabus

CPS 271 – Object-Oriented Programming in C++

Instructor: Khaled Mansour   
Phone: (734) 677-5113

Email: kmansour@wccnet.edu

Office: BE 238   
Office Hours: T 12-1:30, 5-5:30, W 10-11, R 9:30-10,12-1, F 11:30-12

Course Overview

This course is an introduction to Object-Oriented programming using C++ language. Students should have experience using C++ or Java language. Students learn about OOP concepts that are new to traditional languages programmers, Unified Modeling Language, and improved software development processes.

**Pre-Requisite:** CPS 171 (C++) or CPS 161 (Java)

Course Goals

Upon successful completion of this course, you will be able to:

1. Identify appropriate use of Arrays and Dynamic Memory.
2. Identify appropriate uses of objects and classes.
3. Identify appropriate uses of the C++ standard libraries (i.e. string and iostream).
4. Identify appropriate uses of advanced C++ topics.
5. Demonstrate sound software engineering techniques in developing a working software program.

Course Objectives

1. Demonstrate proficiency in processing Arrays.
2. Demonstrate proficiency in pointer manipulation.
3. Demonstrate proficiency in allocating dynamic memory and freeing up memory resources.
4. Demonstrate proficiency using class inheritance.
5. Demonstrate proficiency using constructors and destructors.
6. Demonstrate proficiency using polymorphism.
7. Demonstrate proficiency in using friend functions and classes.
8. Demonstrate proficiency in using operator overloading.
9. Demonstrate proficiency in using the standard string class.
10. Demonstrate proficiency in using the iostream class for text and binary files.
11. Demonstrate proficiency in using C++ exceptions.
12. Demonstrate proficiency in using the various caste operators including dynamic cast.
13. Demonstrate proficiency in using basic templates.
14. Create a program that is logical, easy to understand, and properly indented to compile properly and execute properly to solve a stated problem.

Required Materials

1. Software

For IBM users, you will need Microsoft Visual Studio or any C++ compiler. For Mac users, you will need xcode or eclipse for C++ or any C++ compiler.

* Either the standard edition or the professional edition can be used.
* You will need this software if you wish to do homework on your own computer.
* Before buying this software, make sure you look at the minimum specifications to run it.

Optional Materials

C++ Annotations Version 11.1.1

By: Frank B. Brokken; ISBN: 9036701707

Online: <http://www.icce.rug.nl/documents/cplusplus/>

*The C Programming Language, 2nd Edition*, by Kernighan & Ritchie

<https://www.dipmat.univpm.it/~demeio/public/the_c_programming_language_2.pdf>

*C++ Programming from Problem Analysis to Program Design*, by D.S. Malik

<http://index-of.co.uk/Programming/>

*Object-Oriented Programming in C++, 4th Edition*, by R. Lafore

[https://fac.ksu.edu.sa/sites/default/files/ObjectOrientedProgramminginC4thEdition.pdf](https://nam02.safelinks.protection.outlook.com/?url=https%3A%2F%2Ffac.ksu.edu.sa%2Fsites%2Fdefault%2Ffiles%2FObjectOrientedProgramminginC4thEdition.pdf&data=02%7C01%7C%7Cc8c15421ab59458ff85d08d71832e4c8%7Cec4c99b641114ea0bd5a735a0ae10394%7C0%7C0%7C637004479172614969&sdata=KZaSSeW6j2edxHaT3RmhEGv3GtEIDhBdCMTGcxiA030%3D&reserved=0)

Technical Requirements

* Computer with Internet Access
* (see required materials for more information)

Time Commitment

To meet the due dates on the schedule, expect to spend **12-16** hours per week for this **4** credit course. Designing programs is often time consuming. If you are having difficulty with designing a program, please e-mail me for additional help.

Accommodations

Washtenaw Community College has an open computer lab with the required software. These are located in the Gunder Myran (GM) building above the library. Call 734-973-3420 for hours of operation or visit the [Computer Lab Commons website](http://www.wccnet.edu/resources/computercommons/). Also, study groups are available. Please contact the computer department at 734-677-5431 for the schedule.

If you have a documented disability, contact Learning Support Services at 734-973-3342 as soon as possible to discuss accommodations. Learning Support Services is in room LA 104.

Grading

* Grades are posted in Blackboard under **My Grades**.
* Grading is typically completed within a week of the due date.

Determining Your Grade

Participation

Discussion Boards

Throughout the course there will be discussion boards started for anyone requiring help on the Machine Problems. It is important to use these to help one another work through problems while building your programs. In addition, you will be required to answer questions about the difficulty of the assignment, and to reflect upon challenges/problems you faced. Please see academic integrity for information on what can and cannot be posted. These discussion boards will be graded under participation, you MUST participate in order to receive full points.

Machine Problems

Throughout the course, you will work on seven (7) machine problems. These problems are to help you apply what you have learned in a certain unit. These problems will require you to create a program according to a certain scenario using what you have learned. Each machine problem is worth 50 points and will be due every two weeks.

Quizzes

You will have 4 quizzes throughout your course.

|  |  |  |  |
| --- | --- | --- | --- |
| **Graded Items** | **Number of Assignments** | **Points Each** | **Points Total** |
| **Quizzes** | **4** | **50** | **200** |
| **Machine Problems** | **7** | **50** | **350** |
| **Exams** | **2** | **-** | **450** |
| **Total** | **-** | **-** | **1000** |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Grading Scale** | | | | | | | |
| **A** | 93– 100% | **B** | 82 – 87% | **C** | 72 – 77% | **D** | 62 – 67% |
| **A-** | 90 – 92% | **B-** | 80 – 81% | **C-** | 70 – 71% | **D-** | 60 – 61% |
| **B+** | 88 – 89% | **C+** | 78 – 79% | **D+** | 68 – 69% | **F** | 59% and below |

Fluency in programming cannot be attained by simply reading and studying; you must practice the skills by designing, writing and debugging computer programs on your own. You may get help by emailing me – but there may be some days when I do not check my email so do not expect immediate help this way. My concept of helping you with machine problems is that I will help you learning debugging programs, but I will not debug them for you. Therefore, when you seek help, you should already have some idea about the nature of the program bugs. Make sure to include a copy of your code in your e-mail to speed up the process for me.

Course Deadlines

All assignments will be due on their due dates at 11:59 pm. Please refer to the course schedule for specific due dates and pacing for units.

Late Assignments

No late assignments will be accepted and will receive an automatic zero.

Communication

Email Communication

* + Please send emails from your WCC account to be in compliance with [federal privacy regulations](http://www2.ed.gov/policy/gen/guid/fpco/ferpa/index.html).
  + Provide a clear subject line.
  + Always include your first and last name in the email message.
  + Always include your class and section number in the email message

Expectations and Availability of Instructor

I will usually respond to email within 24 hours Monday-Friday and 48 hours on the weekends and Holidays. You should make a habit of checking your Blackboard course as well as your email on a regular basis as well.

Other Important Information

* An incomplete will only be given for emergencies that arise during the last week of the semester and will not be given for failure to complete assignments when due or to allow a student to complete work to obtain a higher grade.

Academic Integrity

It is always suspect to have machine problem scores significantly higher than your test scores! This usually indicates that you are getting too much help in writing the machine problems. The basic rule is that you may not give or receive assistance for any work that you are submitting as your own.

Some examples of cheating:

* Having someone else write your program (in whole or in part).
* Copying a program someone else wrote (in whole or in part).
* Collaborating with someone else to the extent that the programs are identifiably similar (in whole or in part).

What is not cheating?

* Talking to someone in general about topics and concepts involved.
* Asking someone for help with a specific error message from the compiler. Getting help with the specifics of C++ syntax.
* Using information from the program write-up e.g. copying text describing the problem for your comments.

In addition, please review the material within the Washtenaw Community College Student Policies and Support Information for additional policies and procedures that affect you and your course. Find this information under the Syllabus and Schedule area of this course site.