Course Syllabus

Course Number: CS 203   **Meeting Times:** MWF 1:00 – 1:50 pm
Title: Computer Programming I   **Credit Hours:** 3 Credit Hours
Course Instructor: Marcus Golden   **Course Coordinator:** Timothy Holston
Email: marcus.golden@mvsu.edu   **Office:** 134 CRB & **Phone:** 3401
Office Hours: MW 11 – 12:00pm, **MW** 2 – 3:00pm, **TR** 10 – 12:00pm, **TR** 2:30 - 3:30pm
Website: http://bluebird.mvsu.edu/Faculty/mgolden_profile.html

Catalog Description: This course is an introduction to basic concepts of computer science, with emphasis on object-oriented programming. Fundamental techniques for software design and implementation will be covered and these concepts demonstrated in a programming like C++. Additional topics include top-down modular design, developing general-purpose software tools, procedural and data abstraction and algorithms.

Course Prerequisites: CS 112 or consent of instructor

Textbook(s) and other Required Materials:
T. Gaddis, J. Walters, and G. Muganda, Starting Out with C++: Early Objects, 8th ed., Addison-Wesley, Boston, Massachusetts, 2011.

Program Objectives and Outcomes:
1. Can convey and implement individually or in teams, effective software designs.
   Students will be able to:
   1.1 Demonstrate proficiency in the design and implementation of software in a high-level programming language
3. Have obtained sufficient depth and breadth in computer science, mathematics and science to analyze and solve problems.
   Students will be able to:
   3.1 Use CS, MA and science principles and computing practices to analyze and solve a computer science problem.

Student Learning Outcomes:
Students will be able to:
1. Analyze and explain the behavior of simple programs involving the fundamental programming constructs
2. Design, implement, test, and debug a program that uses each of the following fundamental programming constructs: basic computation, simple I/O, standard conditional and iterative structures, and the definition of functions.
3. Use a design tool to create algorithms for solving simple problems.
4. Understand how precision and round-off can affect numeric calculations.
5. Implement the user-defined data structures in a high-level language
6. Demonstrate the difference between call-by-value and call-by-reference parameter passing.

Student Learning Outcomes Mapped to Program Outcomes												
Computer Programming I												
SLOs	P0 1.1	PO 1.2	PO 1.3	PO 1.4	PO 2.1	PO 2.2	PO 2.3	PO 2.4	PO 3.1	PO 3.2	PO 4.1	PO 4.2
1	x								x			
2	x								x			
3	x											
4	x								x			
5	x											
6	x											

Prerequisite by Topic:
None

Major Topics Covered in Course:	# wks
Computer Science and Computer Languages	1 wk
Algorithms and Problem-Solving Fundamentals	1 wk
Data Types and Output	1 ½ wks
Calculation and Input	1 ½ wks
Software Design	1 wk
Basic Logic	2 wks
Files	1 wk
Selection Statements	1 wks
Repetition Statements	1 ½ wks
Functions	2 wks
Classes	2 ½ wks

Laboratory Projects:
Programming assignments will be required approximately every 1½ - 2 weeks for the major topics listed above.

Estimate ABET Category content		
	Core	Advanced
Data Structures	.5	0
Algorithms	0	0
Computer Organization & Architecture	.5	0
Concepts of Programming Languages	1	0
Software Design	1	0

Oral and Written Communications:
None

Social and Ethical Issues
Issues involving the social and ethical considerations which should be considered in the discipline of computer science are discussed.

Theoretical Content

Data modeling, algorithmic reasoning, and Boolean expressions

Problem Analysis & Solution Design
- Object-oriented analysis and design
- Analysis of simple problems to facilitate development of computer programs to solve problems
- Use of O-O design to develop solutions to simple problems

Assessment
The Final grade will be computed as follows:
Quizzes and tests   20%
Projects and Assignments   25%
Examinations   50%
Participation/Attendance   5%

Grading System
90 - 100   A
80 - 89   B
70 - 79   C
60 - 69   D
59 - below   F

Homework/Lab Assignments: All assignments must be completed and handed in on time at the beginning of class. Make sure you following the format guidelines required for each assignment. I will not accept a partially completed assignment. Also, 10 points will be deducted for each class day that work is turned in late. After 3 days being late, work will not be accepted. Exceptional circumstances should be discussed with the instructor in advance.

Attendance Policy: Attendance is required. Please make every effort to arrive on time! After 5 minutes of the schedule start time of the class, the door will be shut and you will not be allow to enter, thus receiving an unexcused absent for that day. The class roll will be recorded/documented every class session. Sign the roll with you signature. Do not print and do not sign another student's name. I count the number of students every class period. For every 4 unexcused absents, you will receive a letter grade drop. In an event of an absent, you must provide valid documentation base on the university's requirements in order for the absent to not count against you. Students who wish to discontinue the class should officially drop the course; otherwise a grade of "F" will be recorded.

Student Code of Conduct/Civility: Full details may be obtained from the Student Handbook. At a minimum, I expect you to treat each other (and your instructor) politely and with respect. This includes turning off all cell phones (or muting them), participating in class, and arriving in a timely manner. Cell phones/iPods/multimedia devices are to remain in your bags or pockets and should not be visible at any time during the class/lecture/lab hours. No laptop use in the classroom during class hours unless instructed to use them. Please remember that personal conversations during lecture and lab time are distracting to your fellow students and instructor, thus they are not permitted. Collaboration on a project is an exception, of course.

Cheating, Plagiarism/Academic Integrity and Penalties
Any student who submits another student's work as their own will have committed the act of plagiarism. This includes programming assignments and papers. Cutting and pasting from another paper (from web) without giving proper credit to the author of the original paper will be considered plagiarism. Copying parts of another student's paper and programming assignments is also considered plagiarism. The student receives an automatic F on that paper/assignment if it is plagiarized. If the student commits the act of plagiarism a second time, then the student will receive an F grade for that class.

Student with Special Needs: Mississippi Valley State University is committed to providing reasonable accommodations for students with a documented disability. If you feel you are eligible to receive accommodations for a covered disability (medical, physical, psychiatric, learning, vision, hearing, etc.) and would like to request it for this course, you must be registered with the Services for Students with Disabilities (SSD) program administered by University College. It is recommended that you visit the Disabilities Office located inside the EMAP Computer Lab in the Technical Education (IT) Building to register for the program at the beginning of each semester.

For more information or to schedule an appointment, please contact Mr. Billy Benson, Jr. via phone or email at 662-254-3005 or billy.benson@mvsu.edu.

I reserve the right to make changes on this syllabus as needed. This document does not constitute a contract with the University. It contains guidelines.

References

Venit, Stewart and Elizabeth Drake Prelude to Programming Concepts and Design*, Pearson*

Dann, Wanda P., Stephen Cooper and Randy Pausch, *Learning to Program with Alice-Brief Edition*, Pearson/Prentice Hall