

CS 110 – COMPUTER SCIENCE I

College of Information Technology & Engineering
Weisberg Division of Engineering and Computer Science

Semester and Year:

Spring, 2009

Classroom Sections, Locations, and Meeting Times:

Section: 201 CRN: 2100 Days: MWF Time: 9:00 – 9:50 Location: GH211

Textbook:

Horstmann, Cay, *Java Concepts, 5th Ed*, John Wiley & Sons, Inc., 2008 ISBN 978-0-470-10555-9

Course Description:

Object-oriented and algorithmic problem solving principles and techniques; programming with classes in an integrated programming environment; and program debugging.

Pre-requisites:

Students must be either a Computer Science Major or a Pre-Computer Science major, or have a Math ACT score of at least 23. In addition, students must be taking (or have taken) (MTH 127 and MTH 122) or (MTH 130 and MTH 122) or MTH 132 or MTH 229 or MTH 229H

Instructors:

J. Joe Fuller

Office Location: Gullickson Hall Room 205B

Office Hours: MWF 11:00 to 12:00, M 2:00 – 4:00, T 8:30 – 11:00, and W 2:00 – 3:00 (or by appointment)

Phone: (304) 696-6204

Departmental Fax: (304) 696-5454

Email: fullerj@marshall.edu

Jonathan F. Thompson

Office Location: Gullickson Hall Room 205C

Office Hours: MWF 12:00 to 1:00 and MW 2:00 to 3:00 (or by appointment)

Phone: (304) 696-6349

Departmental Fax: (304) 696-5454

Email: thompsonj@marshall.edu

Course Outcome Objectives:

At the completion of this course the student will be able to:

1. Demonstrate competence in a basic programming with classes
2. Explain the various operators available in the programming language, the notions of operator precedence and associativity, and their significance in expression evaluation
3. Evaluate various types of expressions and distinguish between relational, Boolean, arithmetic, and compound expressions
4. Recognize the distinction between syntax, logical, and runtime errors
5. Have a working knowledge of an integrated programming environment and effectively use online documentation
6. Debug programs using a GUI debugger
7. Demonstrate programming techniques for implementing relationships between classes
8. Have a clear understanding of the syntax and semantics of the three fundamental control structures; demonstrate how to choose the right control structure for a given context

Course Activities:*In-class Quizzes*

There will be one quiz for each chapter that we cover in the textbook this semester. Approximately twenty sample quiz questions regarding chapter material will be handed out at least a week in advance of the quiz. The student is expected to complete the chapter reading assignment and answer the sample questions as they go. The in-class closed-book, closed-notes quiz will consist of six questions similar to the sample questions.

Programming Assignments

This course activity involves preparing programs that implement specifications provided by the instructors. Each assignment is designed to use several concepts covered in the course. Students are expected to complete these assignments outside of class and submit them on or before their due date.

Interim Examinations

There will be two interim exams during this semester. Only authorized absences, with prior approval by the instructors, will be accepted for make-up examinations. The first part of the exams will be closed-book and closed-notes. The second part of the exam will be a programming problem that is open-book and open-notes. You're encouraged to use any of your collected works when crafting a solution to the programming problem.

Lab Assignments

Learning to program requires hands-on work at the computer, not just listening to lectures. Hence, we've designed a series of lab exercises that implement the concepts covered in the lectures. We start these labs during class so that the instructors are available to answer questions, clarify the requirements, or to help get you started. If you do not complete a lab during class hours, then you are expected to complete the assignment outside of class on or before the assigned due date. The computer science lab in Gullickson Hall Room 206A is generally available during the day Monday through Friday from 9:00 to 4:00. The lab is also generally open between 4:00 and 9:00pm, Monday through Thursday. Check the schedule posted next to the lab door for available times.

Final Exam

There will be the usual comprehensive two-hour final exam. The format will be the same as the Interim Examinations, that is, part of the exam will be closed book and closed notes and the second part will be a programming assignment that is open-book and open-notes.

Class Attendance, Participation, and Decorum

Students are expected to maintain a certain level of decorum, including:

1. Turning off cell phones and pagers
2. Closing laptops during lectures and class discussions
3. Arriving to class on time
4. Not sleeping during class
5. Not eating during class
6. Keeping side conversations to a minimum
7. Not leaving class until dismissed by the instructors

Evaluation/Grade Computation:

Course grades are based on weighted percentage averages. Your final grade will be derived by multiplying each individual Student Activity score by the weighted percentage and summing all of the weighted percentage averages.

Student Activity	Individual Score	Weighted %	Weighted % Average
In-class quizzes		x 0.15	
Programming Assignments		x 0.15	
Exam 1		x 0.15	
Exam 2		x 0.15	
Lab Exercises		x 0.10	
Final Exam		x 0.20	
Class Attendance, Participation, and Decorum		x 0.10	
Grand Total =			

Evaluation Scale				
90% & Above =A	80% - 89% = B	70% - 79% = C	60% - 69% = D	59% & Below = F

Schedule of Topics:

Week	Mon	Wed	Fri	
1	12-Jan	Course Introduction	Introduction	
2	19-Jan	MLK Birthday (no class)	Using Objects	
3	26-Jan		Implementing Classes	
4	02-Feb			
5	09-Feb	Exam 1 Review	Exam 1	
6	16-Feb		Data Types	
7	23-Feb		Decisions	
8	02-Mar		Iteration	
9	09-Mar			
10	16-Mar	Exam 2 Review	Exam 2	
	23-Mar	University Closed - Spring Break (no class)		
11	30-Mar		Arrays	
12	06-Apr			
13	13-Apr	Designing Classes		
14	20-Apr		Ethics	
15	27-Apr	Machine Language	Exam Review	
	04-May	Exam Week		

Exam Attendance

Students are required to take exams at the scheduled class period. Students may take an exam at a different time under one of the following conditions:

- They present a University Excused Absence
- They present a valid medical excuse
- Other extraordinary circumstance as determined by the instructor

Academic Conduct:

Learning about programming is a hands-on activity, not something that you can pick up by just reading a book or listening to a lecture. It is important that you do the work yourself to gain this experience. To that end, you may discuss programming concepts and techniques with others, consult the web or other textbooks, or study code that is available from various sources but the work you submit must be your own. Here are some examples of appropriate and inappropriate conduct:

- You need to insert an IF statement in your program and you can't remember whether or not parentheses are required. You ask a friend who says, yes, they are required. This is acceptable.
- You're running late on an assignment and in order to hand a program in on time, you copy ten lines of code from a classmate. This conduct is NOT ACCEPTABLE by either student. You must neither directly copy code from someone else nor offer your code to another student or allow it to be copied.
- After struggling for some time, you do a search on the internet and find a snippet of code that you adapt to your problem and insert into your program. You comment your code to acknowledge the source. This is acceptable.
- After struggling for some time, you do a search on the internet and find a program that does exactly what you need. You submit it as your own work. This conduct is NOT ACCEPTABLE.

It is your responsibility to satisfy the spirit of this conduct. If you have any questions, please ask one of the instructors for clarification. Depending on the severity of the offense, the instructors may:

- Take no action
- Penalize the student on the assignment in question
- Assign the student a failing grade in the course

Communication:

Assignments, lecture notes, gradebook, etc, are all posted on MUOnline. Be sure to log in periodically to check for course news. In addition, each student has a folder in the \\CITECS\CS110 file share for program code submissions.

Adaptive Methods for Disabilities:

Students with disabilities who believe that they may need accommodations in this class are encouraged to contact the instructors as soon as possible to better ensure that such accommodations are implemented in a timely fashion. A reasonable period of time must be given to the instructors when making your initial request for any accommodation. Students seeking special accommodations due to disability need to follow the university policy outlined at <http://www.marshall.edu/disabled>.

Bibliography:

Flanagan, David, *Java in a Nutshell, 5th Ed.* O'Reilly & Associates, Inc, Sebastopol, CA 95472

Web Resources:

Java software and documentation	java.sun.com
TextPad text editor	www.textpad.com
BlueJ Development Environment	www.bluej.org
NetBeans Development Environment	www.netbeans.org
Eclipse Development Environment	www.eclipse.org