

CS 110 – COMPUTER SCIENCE I

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

Semester and Year:

Fall, 2007

Classroom Sections, Locations, and Meeting Times:

Section: 101	CRN: 2099	Days: MWF	Time: 9:00 – 9:50	Location: GH211
Section: 102	CRN: 2100	Days: MWF	Time: 1:00 – 1:50	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:

MTH 122 Concurrent or MTH 229 Concurrent or MTH 229H Concurrent or MTH 230 Concurrent

Instructors:

J. Joe Fuller

Office Location: Gullickson Hall Room 205B

Phone: (304) 696-6204

Departmental Fax: (304) 696-5454

Email: fullerj@marshall.edu

Jonathan F. Thompson

Office Location: Gullickson Hall Room 205C

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 basic programming with classes
2. Explain the various operators available in the programming language, understand the notions of operator precedence and associativity, and their significance in expression evaluation
3. Evaluate various types of expressions and distinguishes 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; demonstrates 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 will cover in the textbook this semester. Approximately twenty possible quiz questions regarding chapter material will be handed out a week in advance of the quiz. The student is expected to complete the chapter reading assignment and answer the quiz questions as they go to prepare for the quiz. Six of the twenty questions will appear on the in-class closed-book, closed-notes quiz.

Programming Assignment

This course activity involves preparing an application that implements a specification provided by the instructors. The assignment is designed to use all of the concepts covered in the course. Students are expected to complete the assignment outside of class and submit it on or before the 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 in-class 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. 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 open between 4:00 and 9:00pm, Monday through Thursday. Tutors are available during this time if you need assistance or run into problems.

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 it is 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 Assignment		x 0.10	
Exam 1		x 0.15	
Exam 2		x 0.15	
Lab Assignments		x 0.15	
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	20-Aug	Course Introduction	Introduction to Computing
2	27-Aug	Introduction to Classes	
3	3-Sep	Labor Day (no class)	Data Types
4	10-Sep		
5	17-Sep	Exam 1 Review	Exam 1
6	24-Sep		Decisions
7	1-Oct		Iteration
8	8-Oct	Arrays	
9	15-Oct		
10	22-Oct	Exam 2 Review	Exam 2
11	29-Oct		Designing Classes
12	5-Nov		Java Applets
13	12-Nov		Graphics Programming
	19-Nov	Thanksgiving Break (no class)	
14	26-Nov	Ethics	Machine Language
15	3-Dec	Last Class	Final Review

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, class communications (e-mail), etc, are all handled via the course WebCT/Vista site. Be sure to log into WebCT/Vista regularly to check for course news.

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.

Bibliography:

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

Internet Web Sites:

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