

Wright State University

CORE Scholar

Computer Science & Engineering Syllabi

College of Engineering & Computer Science

Winter 2011

CS 208-01: Computer Programming for Business I

David M. Hutchison

Wright State University - Main Campus, david.hutchison@wright.edu

Follow this and additional works at: https://corescholar.libraries.wright.edu/cecs_syllabi



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

Repository Citation

Hutchison, D. M. (2011). CS 208-01: Computer Programming for Business I. .

https://corescholar.libraries.wright.edu/cecs_syllabi/826

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.

Syllabus

Course Information

Course title: Computer Programming for Business I (4.0 credit hours)
Course number: CS 208 Section 01, Lab Section 01
Course discipline: Computing Sciences
Course description: CS 208 is the first in a sequence of two programming classes required for MIS majors. This course will introduce students to the basic concepts of programming. Examples are from business applications and display graphics and emphasis is on problem solving with the computer as a tool.
Course date: Wednesday, January 5, 2011 through Wednesday, March 16, 2011
Location: 355 Russ Engineering Center
Meeting day(s): Wednesday
Meeting time(s): 6:05 - 9:35 (includes lecture and lab)
Prerequisite(s): Undergraduate level CS 205 Minimum Grade of D and (Undergraduate level MTH 128 Minimum Grade of D or Undergraduate level MTH 129 Minimum Grade of D or WSU Math Placement 05). Familiarity with basic computer concepts and the Microsoft Windows environment is assumed.

WSU Catalog Description

Course description: Introduces basic concepts of programming. Examples are from business applications and display graphics. Emphasis is on problem solving with the computer as a tool.

Instructor Information

Name: David M. Hutchison
Email: david.hutchison@wright.edu
Office location: Inside 1st Floor Russ Lounge
Office hours: Before and after class, as needed or by appointment
Biography:

- B.S. Computer Science, Wright State University, 1990
- M.B.A. Project Management, Wright State University, 1995
- Sun certified Java Programmer
- Sun certified Java Developer
- Project Management Professional
- Current (Winter 2008-Present) instructor of CS 208 and CS 209
- Previous (1997-2001) and current instructor of CS 208 and CS 209
- Instructor of various independent Java, C programming, and Oracle-related Java courses

Teaching assistants: Pavan Kapanipathi Bangalore (kapanipathi.2@wright.edu)

University and Department Requirements

Pre/Co-requisites: Undergraduate level CS 205 Minimum Grade of D and (Undergraduate level MTH 128 Minimum Grade of D or Undergraduate level MTH 129 Minimum Grade of D or WSU Math Placement 05). Proficiency in basic computer usage and Microsoft Windows is assumed.
Program Requirements: BS MIS: Required; Minor in MIS: Elective

Course Requirements

Requirements: This course assumes no prior programming experience. You are required to have a thumb drive or similar media. Although not required, a backup is highly recommended as well. Some students use email for backups - whatever works for you, but you need some type of backup!

Course Learning Outcomes Mapped to Program Outcomes

Course goals: At the completion of the course, students will be able to:

- Describe the process of Java program design and development.
- Identify syntax errors, logic errors, and runtime errors in simple Java programs.
- Design a simple Java program that implements programming constructs such as if/else/if statements, while, do/while, and for loops, and utilizes both logical and relational operators.
- Write efficient, well-documented source code using the Java programming language.
- Analyze a problem statement, translate that analysis into an algorithm, and write a Java program that solves that problem.

Course Policies

Introduction: All course policies are subject to change.

Course Schedule

Schedule: The course schedule presented in the WebCT Calendar is *tentative*. We will do our best to adhere to this schedule, but events beyond our control (e.g., weather) may impact the dates and content. I will do my best to not slip any exam dates.

Class Attendance

Policy: It is your responsibility to attend all classes - attendance will not be taken. Since this class is held only once a week, you stand to miss out on a lot of information, quizzes, labs, project hints, etc. if you choose to skip a class. Of course, prior notice of your missing class would be appreciated.

Grading Policy

Policy: Grading is on a straight 10-point increment scale. That is, >90 is an A, 80-89 is a B, 70-79 is a C, 60-69 is a D, and <60 is an F. Weights of exams, assignments, etc. are as follows:

- 20% - Mid-term exam
- 20% - Final exam
- 25% - Programming assignments
- 10% - Quizzes
- 15% - In-class labs
- 10% - Homework

Remember, your grade is weighted - it is *not* a straight points-based computation.

Course Lab

Lab: You must be enrolled in the lab associated with this class (i.e., CS 208 lab, section 01). The WebCT site for the lab will *not be used*.

In-class Lab Assignments

Weekly lab assignments will be provided during each lab session, beginning with the first night of class. These assignments are to be worked on during lab and submitted prior to the end of the night's lab session, regardless of whether or not they are complete. These assignments will consist of straightforward coding problems such as executing an already written program, writing source code for in complete programs, or designing a complete Java application. Each lab will be worth 50 points.

Academic Dishonesty

Policy: Violators will receive an F for the course and will be reported to the university - official university policy will be followed ([Click here for the policy](#)). You are required to work individually on your programming assignments. You are permitted to exchange ideas with your peers, but you are not permitted to use someone else's work. Additionally, you may not share your work with someone else. If you choose to violate these rules, then all students involved will suffer the consequences.

Course Exams

Exams: All students are required to take both exams. Make-up exams are only given on a case-by-case basis. If you are unable to attend an exam, you are required to provide an acceptable and documented reason *prior* to the exam.

Course Quizzes

Quizzes: Five announced quizzes will be given during the quarter. Makeups will not be allowed without advanced coordination with the instructor (and no one other than the instructor).

Course Homework

Homework: Five outside of class homework assignments will be given. These are straightforward assignments meant to take no more than an hour to complete. The idea is to force you to do some programming outside of class.

Programming Assignments

Assignments: There will be three programming assignments over the course of the quarter. Each of these assignments is worth 100 points, each will state the required due date, and each will state the requirements for that assignment (e.g., provide a design, test cases, source code, etc.). You are required to earn at least 60% of the total points for *each individual* assignment. Failure to earn 60% on each assignment will result in *failure of this course!* Failure to reach an average of 75% across all assignments will also result in *failure of this course!* Late assignments will only be accepted for documented reasons, previously arranged with the instructor (i.e., *not* the lab TA). Please see the lab TA for a description of how points will be allocated for each programming assignment.

Textbook

Required reading: *Java Programming: From The Ground Up*, Ralph Bravaco, Shai Simonson, McGraw-Hill, 1 edition (January 22, 2009), 978-0073523354

Midterm Exam

Content: Chapters 1-4.

Final Exam

Date: March 16, 2011 - 8:00pm til 10:00pm

Content: Comprehensive, with a concentration on chapters 4-6. How much of chapter 6 is included in the final exam will depend on how much we can cover during the last week of the quarter. I will do my best to schedule a review session from 7:00 til 8:00 the evening of the final exam.

Chapter 1

- Topics:
- Course overview
 - What is a computer?
 - Programming languages
 - Java overview
 - Software engineering
 - Program development process

Readings: Bravaco, Chapter 1

Chapter 2

- Topics:
- TextPad
 - What is a program made of?
 - Our first Java program
 - Basic input and output
 - Expressions
 - Data types
 - Operators and precedence
 - Mixing data types
 - The + operator and strings
 - Types of errors

Readings: Bravaco, Chapter 2

Chapter 3

- Topics:
- Variables
 - Obtaining data from outside a program
 - Final variables
 - Conversion Between Primitive Data Types
 - Assignment operators
 - Increment and decrement operators
 - Programming style

Readings: Bravaco, Chapter 3

Chapter 4

Topics:

- if statements
- Nested if statements
- if-else statements
- Nested if-else statements
- if-else-if statements
- switch statements

Readings:

Bravaco, Chapter 4

Chapter 5

Topics:

- while loop
- do-while loop
- for loop
- Nested loops
- break and continue keywords

Readings:

Bravaco, Chapter 5

Chapter 6

Topics:

- Java's predefined methods
- Writing your own methods
- Format of a method definition
- Returning a value
- Void methods
- Calling methods
- Passing parameters to a method
- Local variables
- Scope and lifetime

Readings:

Bravaco, Chapter 6