

Fall 2015

CSCI 1581

Zach Booth
University of New Orleans

Follow this and additional works at: <https://scholarworks.uno.edu/syllabi>

This is an older syllabus and should not be used as a substitute for the syllabus for a current semester course.

Recommended Citation

Booth, Zach, "CSCI 1581" (2015). *University of New Orleans Syllabi*. Paper 161.
<https://scholarworks.uno.edu/syllabi/161>

This Syllabus is brought to you for free and open access by ScholarWorks@UNO. It has been accepted for inclusion in University of New Orleans Syllabi by an authorized administrator of ScholarWorks@UNO. For more information, please contact scholarworks@uno.edu.

Software Design and Development I

CSCI 1581 Section 001
Fall Semester 2015

Zach Booth

Office: MATH 321

email: zachb1988@gmail.com (preferred)

Office Hours: M 8AM-11AM, T 3PM-6PM; other times by appointment only. Office Hours will be held in **Math 321**.

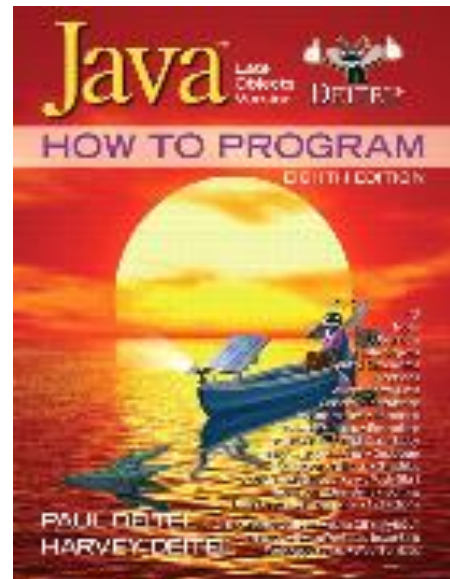
Prerequisite: Math 1115 with a grade of C or better or consent of department; **concurrent registration in CSCI 1583 is required.**

Text: Dietel and Dietel, *Java, How to Program, (Late Objects Version) 8th Ed.*

Course Content:

This lab supports the introductory course CSCI 1583. We will be putting into practice the concepts covered in the lecture. The topics covered (roughly Chapters 1 through 13 and Chapter 16 from the text) will be (we reserve the right to adjust as the term progresses):

- Introduction to Computers and Software Development
- Control Structures and Algorithmic Thinking
- Methods and Structured Programming
- Arrays
- Strings and Files
- Classes and Object-Oriented Programming Concepts
- Inheritance
- Polymorphism
- Exception Handling
- Regular Expressions
- Object-Oriented Design



Laboratory:

The purpose of the lab (CSCI 1581) is to give you an environment to try out concepts in software design via the development of software fragments with a lab

assistant. Attendance and completion of lab work is **mandatory**.

Grading:

(1) Laboratory work (CSCI 1581) will comprise 10% of your final grade. Each lab will consist of a set of typically between 5 and 8 exercises that you will have to submit through Git Lab. The exercises will be equally weighted on a 100 point grading scale. The labs will be graded for completion and correctness.

(2) You will receive the same grade for CSCI 1583 and CSCI 1581. Administrative constraints prevent us from offering the lecture and lab components as a single course. However, they are to be treated as such, hence the single, uniform grade.

(3) All work is graded on a numerical (percentage) basis. The correspondence between numerical and letter grades is given as follows:

A: ≥ 90 ,
B: 80 - 89,
C: 70 - 79,
D: 50 - 69,
F: < 50 .

(4) It is expected that all homework will be turned in on time. Lateness penalties are:

- 1 day late - 10% off;
- 2 days late - 20% off;
- 3 days late - 40% off;
- >3 days late – not accepted

Attendance:

The UNO Senate (Feb. 20, 2002) has made the taking of attendance a requirement for "developmental, 1000, and 2000 level courses." Attendance will therefore be taken at each class meeting. Although not a formal component of the computation of grades, good attendance will impact final grades in borderline cases. Important course content is often introduced outside of the published sources and/or scheduled presentations.

Academic Dishonesty:

Finally, we must call your attention to the University's policies regarding academic dishonesty (<http://www.uno.edu/studentaffairs/accountability.aspx>). Academic dishonesty includes cheating, plagiarism, and collusion. In particular, it includes "the unauthorized collaboration with another person in preparing an academic exercise" and "submitting as one's own any academic exercise prepared totally or in part for/by another." In the event of academic dishonesty, **the student will be assigned a grade of 0** on the exam or exercise, the student will be informed in writing of the action taken, and **a copy of this letter will be**

sent to the Assistant Dean for Special Student Services.

Students with Disabilities:

It is University policy to provide, on a flexible and individualized basis, reasonable accommodations to students who have disabilities that may affect their ability to participate in course activities or to meet course requirements. Students with disabilities are encouraged to contact their instructors and/or the Office of Disability Services to discuss their individual needs for accommodations.

Student Learning Outcomes:

At the conclusion of this course the students will be able to explain what Object-Oriented Programming is and will be able to implement complete computer programs using the object-oriented methodology. Students will be able to identify and explain the various control structures used in programming, will be able to explain what a method is and what happens when a method is called, and will be able to use arrays to process large quantities of data in programs. Students will also be able to apply software design techniques such as inheritance, polymorphism, and exception handling to produce extensible, easily maintainable, and robust programs.

Tentative Schedule of Study:

WEEK 1 (Aug 26-30) Chapters 1 & 2: Introduction

Lab 1: The Linux Command Line and Version Control with Git

WEEK 2 (Sep 2-6) Chapter 3: Control Structures Part I

Lab 2: (Optional) Using the Vim editor

WEEK 3 (Sep 9-13) Chapter 4: Control Structures Part II

Lab 3: Control Structures Part I

WEEK 4 (Sep 16-20) Chapter 5: Methods

Lab 4: Control Structures Part II

WEEK 5 (Sep 23-27) Chapter 6: Arrays

Lab 5: Methods

WEEK 6 (Sep 30- Oct 4) Chapter 6: Strings and Files

Lab 6: Arrays

WEEK 7 (Oct 7-11) MID TERMS

Lab 7: Review for Mid Term

WEEK 8 (Oct 14-18) Chapter 7: Introduction to Classes and Objects

Lab 8: Strings and Files

WEEK 9 (Oct 21-25) Chapter 8: Classes and Objects a Deeper Look

Lab 9: Classes and Objects Part I

WEEK 10 (Oct 28-Nov 1) Chapter 9: Inheritance

Lab 10: Classes and Objects Part II

WEEK 11 (Nov 4-8) Chapter 10: Polymorphism

Lab 11: Inheritance

WEEK 12 (Nov 11-15) Chapter 11: Exceptions

Lab 12: Polymorphism

WEEK 13 (Nov 18-22) Chapter 16: Strings and Regular Expressions

Lab 13: Exceptions

WEEK 14 (Nov 25-29) Chapter 12: Object-Oriented Design with UML

Lab 14: Strings and Regular Expressions

WEEK 15 (Dec 2-6) Chapter 13: Implementing OO Design

Lab 15: Introduction to the Awk Language

WEEK 16 (Dec 9-13) FINALS

FINAL EXAM: Monday, Dec 9th, 8PM - 10PM