

CMSC 150 Introduction to Computing Syllabus

Fall 2016

Course Information

Instructor Information

Instructor: Jory Denny
Email: jdenny@richmond.edu
url: <http://www.mathcs.richmond.edu/~jdenny>
Office: Jepson 226
Office Hours: TWTh 10:30am-12:00pm; Other times by appointment

Lab Assistant (150-03): Thomas Dennis
Email: thomas.dennis@richmond.edu

Lab Assistant (150-04): Emily Ragan
Email: emily.ragan@richmond.edu

Section Information

CMSC 150-03

Lecture:	TTh	9:00am-10:15am	Jepson G30
Lab:	W	1:30pm-3:30pm	Jepson G30
Final:	M Dec. 5	2:00pm-5:00pm	Jepson G30

CMSC 150-04

Lecture:	TTh	1:30pm-2:45am	Jepson G30
Lab:	F	10:00am-12:00pm	Jepson G30
Final	T Dec. 13	9:00am-12:00pm	Jepson G30

Textbook

Introduction to Programming in Java: An Interdisciplinary Approach, Robert Sedgewick and Kevin Wayne,
Pearson Addison Wesley, First Edition, 2007.

<http://introcs.cs.princeton.edu/java/home/>

Website

<http://www.mathcs.richmond.edu/~jdenny/Courses/150>

Course Outcomes

After taking this course a student will be able to:

- Define computer science,
- Diagram basic computer organization, the Java virtual machine, and their relationship,
- Write simple programs in Java to solve application oriented problems using the following features:
 - Primitive data types,
 - String and Math classes of Java,
 - Arrays,
 - Arithmetic and boolean expressions,
 - Control flow (if/else, for, while, etc.),
 - Methods,
 - Input/output, and
 - Simple classes,
- Understand treatment of scope, parameter passing, and data (primitive and user-defined) in Java, and
- Write programs in Java to perform basic searching and sorting of data in arrays including Linear Search, Binary Search, Insertion Sort, and Merge Sort.

Course Content and ***Tentative*** Schedule

During the semester we will discuss the following topics:

Week	Topic	Reading
1	Introduction to Computing, basic programming	Ch 1.1
1, 2	Primitive data, String, Math, expressions	Ch 1.2
3, 4	Control Flow	Ch 1.3
5	Arrays, Java, Review of concepts	Ch 1.4
6, 7	Input/Ouput, Case Study 1	Ch 1.5-1.6
8	Fall break; Midterm	
9, 10, 11	Methods, Recursion, Case Study 2	Ch 2.1-2.4
12, 13, 14	Data types, File I/O, Encapsulation, Inheritance	Ch 3.1-3.4
15	Algorithms for Searching and Sorting	Ch. 4.1-4.2

Note the schedule is subject to change.

Assignments and Grading

All assignments will be announced in class and posted on the course web page. If you miss class for any reason, it is *your* responsibility to find out what assignments you missed.

All programming projects and journal must be submitted in both hard and soft copies.

Your grade will be based on six components:

1. **Exams** — **40%** — There will be one mid-term exam (Oct. 12-14th) and one final exam, each worth 20%. Each exam will have a written and programming component.
2. **Quizzes** — **5%** — There will be a weekly in-class quiz.
3. **Labs** — **5%** — These assignments are to be completed in lab sections, approximately one assignment per week.
4. **Programming Assignments** — **20%** — There will be ten out-of-class programming assignments. These will be turned in through a hard and soft copy.
5. **Programming Project** — **20%** — There will be one out-of-class programming project near the end of the semester spanning multiple weeks. This will be turned in through a hard and soft copy.
6. **Culture Assignment** — **10%** — Each student is to keep a weekly journal through the semester (about 15 in total). This is to give the student the opportunity to explore extra topics and computing interests. Each week the student will write a half page of material on a topic of their choosing related to computer science.

All assignment descriptions are to be found on the course webpage.

Final grades will be assigned according to the following scale:

Percentage	100–90%	89–80%	79–70%	69–60%	60 and below
Final Grade	A	B	C	D	F

Policies

Contact with Lab Assistants

All email contact with lab assistants should CC the primary instructor of the course.

Course Conduct

The student will be respectful to the instructor, lab assistants, and other students. Misconduct will not be tolerated. This includes excessive phone usage, napping, rude commentary, or other disrespectful behavior.

Attendance and Late Assignments

Attendance at all lecture and lab sessions is advised. There will be no make-up exams and no late assignments accepted unless permission from the instructor is sought **in advance**, when possible.

Collaboration

For the assignments in this class, discussion of concepts with others is encouraged, but all assignments must be done on your own, unless otherwise instructed. Reference every source you use, whether it be a person, a book, a paper, a solution set, a web page or whatever. You **MUST** write up your assignments **in your own words**. Copying is strictly forbidden. List all sources for programs in comments at the top of the file and the sources for each journal at the end of your entry.

Academic Integrity

All students are expected to be in accordance with the student honor code. <http://studentdevelopment.richmond.edu/student-handbook/honor/the-honor-code.html>. Note, cheating, lying, plagiarism, academic theft, etc. are not tolerated. If you know another student is breaking the code it is your responsibility to report them to me and the university.

Americans with Disabilities Act (ADA)

The Americans with Disabilities Act (ADA) is a federal antidiscrimination statute that provides comprehensive civil rights protection for persons with disabilities. Among other things, this legislation requires that all students with disabilities be guaranteed a learning environment that provides for reasonable accommodation of their disabilities. If you believe you have a disability requiring an accommodation, please contact the appropriate university coordinator, <http://disability.richmond.edu>.