CMSC 150 Introduction to Computing
Syllabus

Spring 2017

Course Information

Instructor Information

Instructor:	Jory Denny
Email:	jdenny@richmond.edu
url:	`http://www.mathcs.richmond.edu/~jdenny`
Office:	Jepson 226
Office Hours:	TWTh 3:30pm-5:30pm; Other times by appointment

Brief Teaching Philosophy: To state it simply, I believe in learning real world skills and attempting to solve real world problems, i.e., challenges. However, I do not believe in ruining someone's grade because they did not succeed 100%. To facilitate this, we will be using a "flipped" style of class — this implies you should complete reading assignments **BEFORE** coming to class.

Section Information

CMSC 150-04

Lecture:	TTh	9:00am-10:15am	Jepson G30
Lab:	W	10:30pm-12:30pm	Jepson G30
Final:	W Apr. 26	2:00pm-5:00pm	Jepson G30

Requirements

Prerequisite: Basic knowledge of mathematics, algebra, and geometry

Textbook

Required:	*Introduction to Java Programming, Brief Version*, Y. Daniel Liang, Pearson, Tenth Edition, 2015.
Recommended:	*Introduction to Java Programming, Comprehensive Version*, Y. Daniel Liang, Pearson, Tenth Edition, 2015.
Additional Resources:	`http://www.pearsonhighered.com/liang`

Course 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, 2	Primitive data, expressions, Input, Math, and String	Ch 2, 4
3, 4, 5	Control Flow	Ch 3, 5
5, 6	Methods	Ch 6
6, 7	Review; Midterm	
7	File I/O	Ch 12
8	Recursion	Ch 18
9, 10	Arrays	Ch 7, 8
10	Objects	Ch 9
11	Encapsulation	Ch 10
12	Inheritance and Polymorphism	Ch 11
12	Preview of Advanced Topics	Ch 13
13, 14	Performance; Algorithms for Searching and Sorting	

Note the schedule is subject to change.

Assignments and Grading

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

Your grade will be based on five components:

1. **Quizzes — 10%** — There will be in-class quizzes over reading material for the course.
2. **Programming Assignments — 20%** — There will be ten out-of-class programming assignments. These will be turned in through a hard and soft copy.
3. **Exams — 40%** — There will be one mid-term exam and one final exam, each worth 20%. The midterm will have both a written and programming component.
4. **Programming Project — 20%** — There will be one out-of-class programming project near the end of the semester spanning multiple weeks.
5. **Culture Assignment — 10%** — Each student is to write a blog through the semester (10 blogs in total). This is to give the student the opportunity to explore extra topics and computing interests.

Final grades will be assigned according to the following scale:

Final Grade	**Percentage (x)**
A^+	$96.5\% \leq x$
A	$92.5\% \leq x < 96.5\%$
A^-	$89.5\% \leq x < 92.5\%$
B^+	$86.5\% \leq x < 89.5\%$
B	$82.5\% \leq x < 86.5\%$
B^-	$79.5\% \leq x < 82.5\%$
C^+	$76.5\% \leq x < 79.5\%$
C	$72.5\% \leq x < 76.5\%$
C^-	$69.5\% \leq x < 72.5\%$
D^+	$66.5\% \leq x < 69.5\%$
D	$62.5\% \leq x < 66.5\%$
D^-	$59.5\% \leq x < 62.5\%$
F	$x < 59.5\%$

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. **No exceptions!**

Collaboration and Using Resources

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, etc., plagiarism is strictly forbidden. **You must write up your assignments in your own words. List all sources for programs inline in comments or in a comment at the top of the file. For blog entries, properly quote words used and cite sources 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.