

General Information

Foothill College, Fall 2018

Course: CS 1B - Object Oriented Programming in Java

Lecture time/location: Tuesdays/Thursdays 6:00 PM to 7:50 PM in [Room 5607](#)

Instructor Contact Information

Mikel Mcdaniel

mcdanielmikel@fhda.edu (Email any time as much you want.)

Office hours: Tuesdays/Thursdays 7:50 PM to 8:20 PM in the same room as our class (and by appointment)

Texts, Readings, Materials

[Liang, Y. Daniel, Introduction to Java Programming, 10th Edition, Prentice Hall, 2014.](#) ISBN-13: 978-0133761313. ISBN-10: 0133761312.

Detailed Course Information

Tentative Course Schedule and Content

The specific content below is tentative, but this course will cover everything in the official [CS 1B course outline](#). There is 1 "hybrid"/online hour per week intended for homework and Canvas Discussion(s). Student participation in "hybrid" hours is mandatory.

There will be programming assignments every week or two.

The midterm will be on online 2018-11-01 and the final will be in person on [Tuesday, 2018-12-11 from 6 PM to 8 PM](#).

Week	Topics	Chapters/Sections
1	Review of Classes and Accessor Methods. Multi-dimensional Arrays.	1 - 10
2	Inheritance and Polymorphism. ArrayList. Brief intro to Collections, PriorityQueue, Map.	11 (Don't read 20-21)
3	File I/O. Exception Handling.	12 (Optional: 17)
4	Abstract Classes and Interfaces. Comparable vs Comparator. Shallow vs Deep Copy.	13 (Optional: 20.5)
5	Intro to JavaFX (you won't be tested). Bits.	14-16
6	Generics. Midterm.	19
7	Recursion.	18
8	Collections. Iterators. Lists. Queues. Sets. Maps.	20-21
9	Threads.	30
10	All those things I hoped to teach by now, but didn't get to.	
11	Final review. Extra topics.	
12	Final	

Programming Assignments (Grade Oven)

All programming assignments will be submitted to gradeoven.com. Grade Oven will compile and run your code, then give you an instant grade and show your ranking compared to other students, though it won't show your names. Details on how to login will be in your email and/or on Canvas.

Grade Information

Grades are based on programming lab assignments (75% of grade) and the midterm exam (12.5% of grade) and final exam (12.5% of grade). There will be a curve.

Tentative Grading Scale

% needed	Grade
96	A+
93	A
90	A-
86	B+
83	B
80	B-
76	C+
70	C
60	D
0	F

Class Policies

Late Policy

Homework may be turned in almost arbitrarily late. Your score will be calculated as the average of your scores before and after the deadline.

Cheating

Don't.

It's a bad way to end a relationship and it's a bad way to end a class. Please discuss all assignments where the whole class can see it, for example on the Canvas forums. Any variation of copying programming lab assignments is prohibited. The assignment must be 100% your own work.

Attendance

Foothill instructors are required to drop all students who do not come to class in person during the first 2 weeks of class.

Academic Support

Opportunities

Opportunities for Foothill Computer Science students are posted at csopportunities.blogspot.com such as announcements for internships, scholarships, free software offers, pertinent public lectures, etc.

STEM Center

The [STEM Center](#), in room 4213, will have CS tutors at various times each day. The STEM Center is also a place on main campus where students without their own computers can do their lab work. Please ask about online computer science tutors and checkout the [STEM Center schedule](#).

Accommodations

To obtain disability-related accommodations, students must contact [Disability Resource Center](#) (DRC) as early as possible in the quarter. To contact DRC, you may visit DRC in [Room 5400](#), email DRC at adaptivelearningdrc@foothill.edu, and/or call DRC at [650-949-7017](tel:650-949-7017) to make an appointment. If you already have an accommodation notification from DRC, please contact me privately to discuss your needs.

Student Learning Outcomes

- A successful student will be able to use the Java environment to define the basic abstract data types (stacks, queues, lists) and iterators of those types to effectively manipulate the data in his or her program.
- A successful student will be able to define and use Java generics to make their data and algorithms work with a variety of data types.
- A successful student will be able to write and debug Java programs which make use of inheritance, i.e., the "is a" relationship, common to all OOP languages. Specifically, the student will define base and derived classes and use common techniques such as method chaining in his or her programs.