

COMSC 075–Summer 2020

Instructor:

J. David Eisenberg (david.eisenberg@evc.edu)

Office Hours:

Online via Zoom or other video conference system

General Information

3 Units | CR

Course Description

An introduction to computer programming concepts from a computer science perspective.

Programming structures of sequential, selection and repetition are covered using a high-level object-oriented language. Using existing classes and creating classes, methods, argument passing, program and data abstraction (including arrays) are covered. Documentation, programming style and program design and development are addressed throughout the course.

Course Prerequisites

MATH 21 and MATH 22 or MATH 25, or equivalent with a grade of “C” or better.

Student Learning Outcomes

Upon completion of this course, students will be able to:

- Design, code, debug, test, and document simple programs that use the fundamental constructs: basic computation and sequencing, decision and iterative structures, and the definition of functions (methods).
- Analyze problem descriptions, apply problem-solving methods, and use design tools to develop algorithms to solve simple problems.
- Identify and describe the properties of a variable such as its associated address, type, value, scope, persistence, and size.
- Develop programs to create and process sequential data files.
- Summarize the history and evolution of programming languages, including paradigms in current use.

Process

This class consists of lecture (mostly via video) and lab--programming exercises which you will do on your own computers..

Required Text

Introduction to Java Programming and Data Structures, 11th Edition, by Y. Daniel Liang. Pearson, 2018 ISBN 978-0-13-467094-2

You may also use the 10th edition of the book.

Assignments

You will be assigned several programming projects ranging from light to moderate complexity. The material you send to the instructor must contain the source code with comments that include your name, the assignment number, and a description of the problem to be solved. Assignments will be graded on functionality, organization, readability (which includes proper use of naming and indentation), and appropriate use of programming constructs.

When you upload an assignment, you *must* name the files exactly as required in the assignment, or they will not be graded. File names may *not* contain blanks unless specifically required.

Exams

You will have two midterms and one final; there may be pop quizzes as well. The comprehensive final will be on the last day of class, 23 July 2020, during normal class hours. The final will cover topics that include, but are not limited to: determining what a program fragment produces as output, completing the writing of a partial program, and debugging a non-working program.

Grading

The work on the tests and exercises will be weighted and combined to calculate your final grade as follows:

Weight		Grading	
Programs:	180 points	450-500	A
Midterms:	200 points	400-449	B
Final Exam:	120 points	350-399	C
Total:	500 points	300-349	D
		Under 300 points	F

Attendance

If you decide to drop the class, it is your responsibility to do so. Five missed assignments may cause you to be dropped (but I will contact you first before doing so)—please notify the instructor if you have any extenuating circumstances. If you know in advance that you won't be able to get work done, please send me an email. Deadline dates for dropping a class with and without a "W" are printed in the EVC Schedule of Classes.

Policies

Exercises are to be submitted prior to the end of the class lab period on the date due. If turned in late, you will receive reduced credit; one letter grade per week late. (After two sessions late, I reserve the option to not grade the assignment.) Missed tests or exercises will receive zero credit. You may request to take a test at a different time, so long as your request is made in advance of the scheduled examination and is submitted in writing. Your request may or may not be granted. Illnesses and emergencies are another thing altogether, and I will make reasonable exceptions in those cases. All examinations must be taken to complete this course.

Discussion about assignments is encouraged, but you must each do your own work. Cheating and plagiarism will be met with an F on the assignment. See the EVC catalog for details on the College Honesty Policy as well as student disciplinary and grievance procedures.

If you have learning or physical needs that require special accommodations, please make an appointment with the Disabled Students Program, and notify me in writing of your needs.

Study Hints

1. Read the assigned materials *before* attending class.
2. Type all the example programs from the text and try them. Don't just copy and paste.
3. Feel free to experiment with extending or modifying example programs from the text to make them do different things.
4. After the class meeting, re-read the assigned materials and review lecture notes.
5. Allow plenty of time for completion of programming assignments.

Early Alert

At EVC, your academic success matters. We want to ensure that you have the tools and support you need to be academically successful, so EVC has implemented an Early Alert program. The goal of the Early Alert program is to support your success by providing intervention and support services. The Early Alert process begins with your instructor identifying you as a student who may be in need of academic support if they see you struggling in their classes for various reasons. Designed not to be punitive or judgmental, Early Alert is not reflected on your transcript and does not affect your financial aid or GPA. [Here are answers to Frequently Asked Questions about Early Alert \(Links to an external site.\)](#).

Course Outline

Note: the pace at which we will cover these subjects may deviate from this printed schedule. Your mileage may vary.

Week	Chapter(s)	Topics	Assignment(s) Due
15 June 2020	Chapters 1, 2, 3	<ul style="list-style-type: none"> • Introduction to Computers and Java Fundamentals • Identifiers, Numeric Data Types, and Operations • Evaluating expressions • The Software Development Process • The boolean data type • Selections with if and switch statements • Program Testing 	Essay Introduction to Java Variables
22 June 2020	Chapters 4, 5, 6	<ul style="list-style-type: none"> • Mathematical Functions • The char and String Data types • Introduction to Loops: <ul style="list-style-type: none"> • do-while loops • for loops • nested loops • void vs. Value-returning Methods • Parameter Passing • Modularization of Code • The Scope of Variables • Method Abstraction and Stepwise Refinement 	Selection Structures
29 June 2020	Chapters 7 and 8	<ul style="list-style-type: none"> • Introduction to Arrays • Programming examples • Copying Arrays • Passing Arrays to Methods • Searching and Sorting • The Arrays Class • Multi-dimensional Arrays 	Strings and Loops Midterm 1
6 July 2020	Chapters 9 and 10	<ul style="list-style-type: none"> • Introductions to Classes and Objects • Using Classes from the Java Library • static Variables and Methods • Passing Objects to Methods • Arrays of Objects • Scope of Variables • Class Abstraction and Enapsulation • Class Relationships • Wrapper Classes • The String and StringBuilder Classes 	Methods One-dimensional Arrays Midterm 2
13 July 2020	Chapter 11	<ul style="list-style-type: none"> • Inheritance • Overriding vs. Overloading • Polymorphism • Dynamic Binding 	Multidimensional Arrays Objects and Classes

Week	Chapter(s)	Topics	Assignment(s) Due
20 July 2020	Chapter 12	<ul style="list-style-type: none"> <li data-bbox="589 128 886 159">• The ArrayList Class <li data-bbox="589 210 878 241">• Exception handling <li data-bbox="589 245 984 275">• Files and Text Input/Output 	Inheritance and Polymorphism Files and Exceptions Final Exam