MT. SAN ANTONIO COLLEGE, WALNUT

Math/Computer Science Department
Spring 2023
Syllabus for CSCI 145 -- 43069
Java Language and Object-Oriented Programming
Prof. Tuan A Vo
M 1:15pm – 4:15pm, Room 61-1418
W 1:15pm – 4:15pm, Online via Zoom

Course Description

For computer science, engineering, mathematics, and other science students. Introduction to Java language and object-oriented programming with Java as well as general concepts and techniques of computer programming. Topics include Java expressions, flow control, functions and program structure, Java classes, overloading, object references, inheritance, Java library packages, exceptions, file I/O, applets, event handling, multithreading. Prerequisite: Completion of CSCI 110. 4 units (3 lecture units and 1 lab unit).

Student Learning Outcomes (SLO)

1. Students will be able to analyze problems and design appropriate algorithms.
2. Students will be able to code provided algorithms using Java language.
3. Students will be able to provide code for a Java class given objects' attributes and behaviors.
4. Students will be able to use existing Java classes to perform required tasks.

Course Measurable Objectives (CMO)

1. Analyze problems and design appropriate algorithms.
2. Code algorithms into the Java language.
3. Recognize and produce proper Java syntax.
4. Utilize recursion, iteration, and arrays.
5. Demonstrate the paradigm of object-oriented programming.
6. Write, organize, and assemble program documentation.
7. Develop standards for comparing the efficiency of various algorithms.
8. Demonstrate debugging techniques.

Office Hours and Addresses

Office Location:	61-1654
Office Hours:	MW 12:30 pm – 1:15 pm (61-1654) Tu 7:15 am – 7:45 am (61-1654) Tu 11:00 am – 11:45 am (61-1654) W 12:30 pm – 1:15 pm (ID: 892 1501 3468, Code: 812348) Th 7:15 am – 7:45 am (ID: 947 5908 4693, Code: 344393) Th 11:00 am – 11:45 am (ID: 916 9238 5594, Code: 539378)
Office Phone:	(909) 274-4519
Email:	tvo@mtsac.edu
Zoom Meeting:	ID: 822 4506 5598, Code: 992290 (Wednesday)
Course Link:	https://myportal.mtsac.edu (via Canvas)

Textbooks and Materials

- J. Lewis and W. Loftus, *Java Software Solutions -- Foundations of Program Design*, 9th Edition, Addison-Wesley, 2017 (can also use 8th edition for reference).
- Access to a computer with a Java compiler (available in Math/CSCI computer lab).

Grading

Grading will be based on homework assignments, pop quizzes/in-class exercises, exams, and programming assignments. The final grade is generally based on a straight scale:
A >= 90%, 80% <= B < 90%, 70% <= C < 80%, 60% <= D < 70%, F < 60%.

Percent	Items	Lecture Points	Lab Points
24%	Programming Assignments -- 12		240
6%	Homework -- 3	60	
6%	Pop Quizzes/In-class Exercises -- 6	60	
36%	Exams -- 2	360	
4%	Lab Final		40
24%	Written Final -- comprehensive	240	

Unless you received my prior approval, late assignments will only be accepted up to the next class session with a 10% deduction. Extra late assignments without my prior approval might be accepted at my discretion with at least 50% deduction. Programming assignments are due at the end of the class on the due date and most other assignments are due at the beginning of the class on the due date. You will typically get your assignments back within one week. All exams will cover the homework assignments, textbook reading assignments, programming assignments, and class activities. The exams will be on campus, and they will usually consist of true/false, multiple choice, problem solving, and short essay questions. Exams can only be made up with my prior approval or special circumstance that can be substantiated in timely manner (i.e., notify me within 24 hours after the exam for special circumstance). For other cases, a makeup exam might be allowed at my discretion with at least 10% deduction.

Some potential extra credit points can be earned from programming assignments, class activities, and excellent work. A pop quiz or in-class exercise can be given any time so make sure to come to class on time and be there for the duration of the class. You can only make-up either one pop quiz or one in-class exercise at the end of the semester.

Lab Activities

The goal of the lab is to enhance teaching/learning and to develop a deeper understanding of concepts taught in class. We will use lab time to lecture on topics, demonstrate new concepts via programming examples, and you would have time to try out new concepts under the supervision of your professor. Lab activities would include going over new concepts, solving problems, modifying existing programs, testing programs, debugging programs, and conducting experiments. Important concepts will be demonstrated so you can complete your programming assignments. Besides coding analysis, design, testing and debugging it will be important that you learn individually or in collaborative group work. You might be asked to present/share your solutions with other students so giving you a chance to learn from others.

Students need to prepare for lab by reading the material assigned and covered in lecture. Each lab will start with a preview of the concepts on which the lab is based in one of the following forms: instructor lecture or demonstration, class questions or discussion, using class worksheets: T/F, fill-in blanks, or short answers. Class lab handouts will guide you to use the lab resources, lab textbook or interactive visualization software. Handouts on specific additional topics, code skeletons or demonstrations will be provided.

As you work on code, it is recommended that you talk to each other (unless otherwise instructed) or ask your instructor for individual help. If a topic will be of interest for the whole class, we will take it up for class discussion and your instructor will lecture or demonstrate to clarify that topic. Your instructor will take questions to make sure you understand the concepts. Programming will be done on mostly MS Windows OS, but we might explore some other OS as well. Your answers to questions on the lab handouts will provide feedback and help access your work. You will be asked to formulate your own conclusions in the worksheets, and we shall use discussions, presentations, or lectures to present/debate these conclusions. The programming assignments and lab final represent 28% of your total grade.

Lab grading:
        Programming Assignments -- 240 points
        Mandatory lab final -- 40 points
        Total lab -- 280 points

Cheating/Plagiarism

Cheating and plagiarism will not be tolerated in any shape or form in this class. Unless you are prepared to do your own work, DO NOT take this class. Zero will be given to an assignment for the first offense and a second offense would result in a grade of "F" for the course. In addition, you may be subject to Mt. San Antonio College's student discipline process. This policy is applied to both the copier and/or the provider so protect your work. You cannot work together on any assignments unless directed by your instructor. Academic dishonesty includes but is not limited to cheating on a test, sharing a solution, sharing code, and using unauthorized code. It is not okay to let others use your

code! When in doubt, ask your instructor. In addition, refer to the school catalog for additional information.

Useful Information

Last day to drop without a W is 03/05/2023 and last day to drop with a W is 04/28/2023. It is your responsibility to drop from the class or you will receive a grade. Since programming is part of the course, be prepared to be very patient and willing to spend a great deal of time with and without a computer! Any Java compiler can be used to complete your assignments.

I usually check email at least once a day during weekday so you can expect a response from me within 24 hours during weekday and it would take longer for the weekend. For immediate help, it would be best to email me during office hours or see me during my office hours via Zoom. For help with a program, make sure to have an electronic copy of your program available.

We will have seating assignment so do come early in the second week to select your seat. Do not eat or drink in the classroom. If you have any special accommodation needs such as ACCESS program, please let your instructor know as soon as possible. Please advise your instructor of any special requirements within the first week. The instructor reserves the right to revise this syllabus.

Zoom lectures will be recorded and they are available to the whole class for later viewing. I also may share the recorded lectures with another section if applicable. If you are concerning about your privacy, you can keep your camera off during the recording and you can change your screen name to a nick name. Please let me know if you are going to use a nick name so I know that you are attending the class. However, if you still have any issues with the lectures being recorded for later viewing or being shared with another section, please contact me.

Tentative Schedule

This is just a tentative schedule and is subject to change. We will cover chapters 1 through 13. Lectures and reading assignments will cover the materials from the required text and it is your responsibility to read the assigned materials before coming to class. Many students tell me they have a better understanding of the materials when they read those topics before the lecture.

WEEK	DATE (M)	LECTURE ACTIVITIES	LAB ACTIVITES (lecture and demonstrate code)	TEXTBOOK
1	02/20	Holiday (M), Introduction, Computer Systems	Software Development Environment, Java Language, and Java Programs	Chapter 1
2	02/27	Data and Expressions	Java Operators and Expressions, Input and Output	Chapter 2
3	03/06	Using Classes and Objects	Java Applications, Graphics	Chapter 3
4	03/13	Writing Classes	Java Classes and User-Defined Classes	Chapter 4
5	03/20	Writing Classes & GUI, HW 1, Review	Components of a Class and GUI Components	Chapters 4 & 5
6	03/27	Conditional and Loops, Exam 1, Holiday (F)	Selection Structure, Nested Selection	Chapter 6
7	04/03	Conditional and Loops	Repetition Structure, Types of Loops, File Input	Chapters 6 & 7
8	04/10	Object-Oriented Design	Software Development Activities, Java Interfaces	Chapters 7 & 8
9	04/17	Arrays	Setting up Arrays and Array Applications	Chapter 8
10	04/24	Inheritance, HW 2, Review	Inheritance and Overriding, Abstract Methods	Chapter 9
11	05/01	Polymorphism, Exam 2	Abstract Classes, Polymorphism with Inheritance and Interface	Chapter 10
12	05/08	Exceptions	Exception Handling, File Output	Chapters 10 & 11
13	05/15	Files, Threads (Supplement)	Multi-threading with Thread and Runnable	Chapter 11
14	05/22	Recursion	Setting up and Tracing Recursive Methods	Chapter 12
15	05/29	Holiday (M), Introduction to Data Structures, HW 3, Review	Using existing data structures	Chapter 13
16	06/05	Written Final – Monday @ 1:30 pm	Lab Final – Wednesday @ 1:30 pm	