I. GENERAL COURSE INFORMATION

Subject and Number: Computer Science 3
Descriptive Title: Object-Oriented Programming in Java
Course Disciplines: Computer Science
Division: Mathematical Sciences

Catalog Description:
This course includes a detailed coverage of object-oriented programming concepts and design using the Java programming language, including Java data types, operators and expressions, control structures, iterations, functions, arrays, classes and inheritance, files, and graphical user interface (GUI) applications with event handling.

Conditions of Enrollment:
Prerequisite : Computer Science 1 with a minimum grade of C or equivalent

Course Length:	**X Full Term**	**Other (Specify number of weeks):**
Hours Lecture:	**3.00 hours per week**	**TBA**
Hours Laboratory:	**3.00 hours per week**	**TBA**
Course Units:	**4.00**	

Grading Method: Letter
Credit Status Associate Degree Credit

Transfer CSU: X Effective Date: 2/20/2001
Transfer UC: X Effective Date: Fall 2002

General Education:

El Camino College:

CSU GE:

IGETC:

II. OUTCOMES AND OBJECTIVES

A. COURSE STUDENT LEARNING OUTCOMES (The course student learning outcomes are listed below, along with a representative assessment method for each. Student learning outcomes are not subject to review, revision or approval by the College Curriculum Committee)

1. Students, when given a specification for a program or program segment, will be able to design, code, compile, test and document a solution.

2. When given a code segment involving control structures, iteration structures and method calls, students will be able to trace the execution and give the output.

3. When given a code segment involving data objects, graphical interface objects, and processing objects, students will be able to describe what the user sees and the events that take place as the user interacts with the interface

4. Students understand and be able to develop both desktop and web applications involving graphical user interfaces, animations, sound manipulations, File I/O, database, applets, and threads.

The above SLOs were the most recent available SLOs at the time of course review. For the most current SLO statements, visit the El Camino College SLO webpage at http://www.elcamino.edu/academics/slo/.

B. Course Student Learning Objectives (The major learning objective for students enrolled in this course are listed below, along with a representative assessment method for each)

1. Identify Java data types.
   - Written homework
2. Develop programming code using control and iteration statements.
   - Laboratory reports
3. Develop programming code using strings and arrays, both single and multidimensional.
   - Laboratory reports
4. Construct and use objects from predefined classes.
   - Objective Exams
5. Write and use static (class) and instance methods.
   - Objective Exams
6. Construct classes to encapsulate data and methods.
   - Objective Exams
7. Demonstrate ability to use inheritance and polymorphism in program code.
   - Laboratory reports
8. Develop programming code using Input/Output files.
   - Laboratory reports
9. Declare and use graphical components for user interfaces; handle basic mouse and keyboard events.
   - Objective Exams
10. Create and use applets with threads in world wide web applications.
    - Laboratory reports

III. OUTLINE OF SUBJECT MATTER (Topics are detailed enough to enable a qualified instructor to determine the major areas that should be covered as well as ensure consistency from instructor to instructor and semester to semester.)

Lecture or Lab	Approximate Hours	Topic Number	Major Topic
Lecture	4	I	INTRODUCTION TO JAVA A. Java programming environment B. Algorithm development C. Fundamental data types
Lecture	4	II	JAVA CONTROL STRUCTURES A. Boolean Expressions and truth tables, short circuit evaluation B. If-else statements C. Nested if/else statements D. Switch statements
Lecture	4	III	JAVA ITERATION INSTRUCTIONS A. For statements B. While statements C. Do-while statements D. Enhanced for loop
Lecture	4	IV	JAVA METHODS A. Variables and scope of variables B. Passing arguments C. Void and value returning methods D. Library classes and their methods
Lecture	10	V	JAVA CLASSES A. Encapsulations of data using private and protected modifiers B. Instance, static, and final class data members C. Direct initialization D. Initialization blocks E. Default constructors F. Explicit Constructors G. Constructor overloading and chaining H. Instance methods as interfaces I. Static Methods J. Public and private methods K. Overloading of methods in general and the ones inherited from object class.
Lecture	6	VI	USING LARGE BLOCKS OF MEMORY OR COLLECTIONS A. One dimensional arrays B. Multidimensional arrays C. Array Lists D. Vectors E. Searching Collections F. Sorting Collections
Lecture	4	VII	FILE INPUT/OUTPUT

			A. Scanner Class use for reading input files B. Print Stream Class use for output to files C. Use of print f function to format output data D. Buffered Streams
Lecture	4	VIII	PACKAGES, SOURCE AND CLASS FILES A. Creating and compiling Java Packages B. Multi-file and multi class programs C. .java source files and .class files D. Debugging Java Programs
Lecture	7	IX	JAVA OBJECT ORIENTED TECHNOLOGY A. Inheritance B. Interfaces C. Multiple inheritance from interfaces D. Polymorphism E. Virtual methods and method overriding F. Exceptions
Lecture	7	X	GRAPHICAL USER INTERFACES (GUIS) A. Event handling B. Applying inheritance and polymorphism principles related to GUI programming C. Inner classes D. Anonymous classes E. Abstract classes F. Adapter classes G. Java Applets
Lab	4	XI	Labs related to the INTRODUCTION TO JAVA A. Java programming environment B. Algorithm development C. Fundamental data types
Lab	4	XII	Labs related to JAVA CONTROL Structures A. Boolean Expressions and truth tables, short circuit evaluation B. If-else statements C. Nested if/else statements D. Switch statements
Lab	4	XIII	Labs related to JAVA ITERATION INSTRUCTIONS A. For statements B. While statements C. Do-while statements D. Enhanced for loop
Lab	4	XIV	Labs related to JAVA METHODS A. Variables and scope of variables B. Passing arguments C. void and value returning methods D. Library classes and their methods
Lab	10	XV	Labs Related to JAVA CLASSES A. Encapsulations of data using private and protected modifiers

			B. Instance, static, and final class data members C. Direct initialization D. Initialization blocks E. Default constructors F. Explicit Constructors G. Constructor overloading and chaining H. Instance methods as interfaces I. Static Methods J. Public and private methods K. Overloading of methods in general and the ones inherited from object class.
Lab	6	XVI	Labs related to USING LARGE BLOCKS OF MEMORY OR COLLECTIONS A. One dimensional arrays B. Multidimensional arrays C. Array Lists D. Vectors E. Searching Collections F. Sorting Collections
Lab	4	XVII	Labs related to FILE INPUT/OUTPUT A. Scanner Class use for reading input files B. Print Stream Class use for output to files C. Use of print f function to format output data D. Buffered Streams
Lab	4	XVIII	Labs related to PACKAGES, SOURCE AND CLASS FILES A. Creating and compiling Java Packages B. Multi-file and multi class programs C. .java source files and .class files D. Debugging Java Programs
Lab	7	XIX	Labs related to JAVA OBJECT ORIENTED TECHNOLOGY A. Inheritance B. Interfaces C. Multiple inheritance from interfaces D. Polymorphism E. Virtual methods and method overriding F. Exceptions
Lab	7	XX	Labs related to GRAPHICAL USER INTERFACES (GUIS) A. Event handling B. Applying inheritance and polymorphism principles related to GUI programming C. Inner classes D. Anonymous classes E. Abstract classes F. Adapter classes G. Java Applets
Total Lecture Hours		54	
Total Laboratory Hours		54	
Total Hours		108	

IV. PRIMARY METHOD OF EVALUATION AND SAMPLE ASSIGNMENTS

A. **PRIMARY METHOD OF EVALUATION:**
Problem solving demonstrations (computational or non-computational)

B. **TYPICAL ASSIGNMENT USING PRIMARY METHOD OF EVALUATION:**
Write an applet that draws a filled green circle at the pointer location each time the mouse is clicked. First, use an ACTION() handler to trap the mouse click. Next, use a MOUSEUP() handler.

C. **COLLEGE-LEVEL CRITICAL THINKING ASSIGNMENTS:**
1. Develop an algorithm to read the following input list of test scores, sort them in ascending order and print the list. Translate the algorithm into Java code and test the program for correctness.

2. A prime number is an integer greater than 1 and divisible only by itself and one. The first seven prime numbers are 2, 3, 5, 7, 11, 13 and 17. Write an application to include a method that prints out all factors of the argument, and another method to determine if the argument is a prime number.

D. **OTHER TYPICAL ASSESSMENT AND EVALUATION METHODS:**
Other exams
Quizzes
Homework Problems
Multiple Choice
Completion
Matching Items
True/False
Other (specify):
Computer programs written to specifications
Design of a computer assignment
Write, Test and Run Programs that meet Assignment Specifications Free response to program design questions

V. INSTRUCTIONAL METHODS
Demonstration
Discussion
Guest Speakers
Laboratory
Lecture

Note: In compliance with Board Policies 1600 and 3410, Title 5 California Code of Regulations, the Rehabilitation Act of 1973, and Sections 504 and 508 of the Americans with Disabilities Act, instruction delivery shall provide access, full inclusion, and effective communication for students with disabilities.

VI. WORK OUTSIDE OF CLASS
Study
Required reading
Problem solving activities
Other (specify)
Writing computer programs

Estimated Independent Study Hours per Week: 6

VII. TEXTS AND MATERIALS

A. UP-TO-DATE REPRESENTATIVE TEXTBOOKS

Lewis and Loftus. Java Software Solutions: Foundations of Programming Design. 9th ed. Pearson, 2018.

B. ALTERNATIVE TEXTBOOKS

C. REQUIRED SUPPLEMENTARY READINGS

D. OTHER REQUIRED MATERIALS

VIII. CONDITIONS OF ENROLLMENT

A. Requisites (Course and Non-Course Prerequisites)

Requisites	Category and Justification
Course Prerequisite Computer Science-1 or	Sequential
Non-Course Prerequisite	If students have not taken CSCI 1 but have taken a similar course at another college or have programming experience at work, students will have the skills needed to take this course. It is recommended that students have basic programming skills or they may not succeed in this course. The student will be exempted from the prerequisite if she/he can demonstrate sufficient programming knowledge through work portfolio or oral and/or written examination by the department of computer science faculty.

B. Requisite Skills

Requisite Skills
1. Develop algorithms using a programming language such as C++.
2. Use one-dimensional arrays within a programming language such as C++.
3. Develop subprograms using a programming language such as C++.

C. Recommended Preparations (Course and Non-Course)

Recommended Preparation	Category and Justification

D. Recommended Skills

Recommended Skills

E. Enrollment Limitations

Enrollment Limitations and Category	Enrollment Limitations Impact

Course created by Massoud Ghyamkhah on 04/01/1999.

BOARD APPROVAL DATE: 06/21/1999

LAST BOARD APPROVAL DATE: 12/16/2019

Last Reviewed and/or Revised by Edwin Ambrosio on 9/9/2019
20157