

Tick one box per statement: ☐ not yet ☐ nearly ☐ confident. / /

1. Using Objects and Methods

1.1 Introduction to Algorithms, Programming, and Compilers

- ☐☐☐ Represent patterns and algorithms found in everyday life using written language or diagrams.
- ☐☐☐ Explain the code compilation and execution process.
- ☐☐☐ Identify types of programming errors.

1.2 Variables and Data Types

- ☐☐☐ Identify the most appropriate data type category for a particular specification.
- ☐☐☐ Develop code to declare variables to store numbers and Boolean values.

1.3 Expressions and Output

- ☐☐☐ Develop code to generate output and determine the result that would be displayed.
- ☐☐☐ Develop code to utilize string literals and determine the result of using string literals.
- ☐☐☐ Develop code for arithmetic expressions and determine the result of these expressions.

1.4 Assignment Statements and Input

- ☐☐☐ Develop code for assignment statements with expressions and determine the value that is stored in the variable as a result of these statements.
- ☐☐☐ Develop code to read input.

1.5 Casting and Range of Variables

- ☐☐☐ Develop code to cast primitive values to different primitive types in arithmetic expressions and determine the value that is produced as a result.
- ☐☐☐ Describe conditions when an integer expression evaluates to a value out of range.
- ☐☐☐ Describe conditions that limit accuracy of expressions.

1.6 Compound Assignment Operators

- ☐☐☐ Develop code for assignment statements with compound assignment operators and determine the value that is stored in the variable as a result.

1.7 Application Program Interface (API) and Libraries (API)

- ☐☐☐ Identify the attributes and behaviors of a class found in the libraries contained in an API.

1.8 Documentation with Comments

- ☐☐☐ Describe the functionality and use of code through comments.

1.9 Method Signatures

- ☐☐☐ Identify the correct method to call based on documentation and method signatures.

- ☐☐☐ Describe how to call methods.

1.10 Calling Class Methods

- ☐☐☐ Develop code to call class methods and determine the result of those calls.

1.11 Math Class Math

- ☐☐☐ Develop code to write expressions that incorporate calls to built-in mathematical libraries and determine the value that is produced as a result.

1.12 Objects: Instances of Classes

- ☐☐☐ Explain the relationship between a class and an object.
- ☐☐☐ Develop code to declare variables to store reference types.

1.13 Object Creation and Storage (Instantiation) ()

- ☐☐☐ Identify, using its signature, the correct constructor being called.
- ☐☐☐ Develop code to declare variables of the correct types to hold object references.
- ☐☐☐ Develop code to create an object by calling a constructor.

1.14 Calling Instance Methods

- ☐☐☐ Develop code to call instance methods and determine the result of these calls.

1.15 String Manipulation

- ☐☐☐ Develop code to create string objects and determine the result of creating and combining strings.
- ☐☐☐ Develop code to call methods on string objects and determine the result of calling these methods.

2. Selection and Iteration

2.1 Algorithms with Selection and Repetition

- ☐☐☐ Represent patterns and algorithms that involve selection and repetition found in everyday life using written language or diagrams.

2.2 Boolean Expressions

- ☐☐☐ Develop code to create Boolean expressions with relational operators and determine the result of these expressions.

2.3 if Statements if

- ☐☐☐ Develop code to represent branching logical processes by using selection statements and determine the result of these processes.

2.4 Nested if Statements if

- ☐☐☐ Develop code to represent nested branching logical processes and determine the result of these processes.

2.5 Compound Boolean Expressions

- ☐☐☐ Develop code to represent compound Boolean expressions and determine the result of these expressions.

2.6 Comparing Boolean Expressions

- ☐☐☐ Compare equivalent Boolean expressions.
- ☐☐☐ Develop code to compare object references using Boolean expressions and determine the result of these expressions.

2.7 while Loops while

- ☐☐☐ Identify when an iterative process is required to achieve a desired result.
- ☐☐☐ Develop code to represent iterative processes using ‘while’ loops and determine the result of these processes.

2.8 for Loops for

- ☐☐☐ Develop code to represent iterative processes using ‘for’ loops and determine the result of these processes.

2.9 Implementing Selection and Iteration Algorithms

- ☐☐☐ Develop code for standard and original algorithms (without data structures) and determine the result of these algorithms.

2.10 Implementing String Algorithms

- ☐☐☐ Develop code for standard and original algorithms that involve strings and determine the result of these algorithms.

2.11 Nested Iteration

- ☐☐☐ Develop code to represent nested iterative processes and determine the result of these processes.

2.12 Informal Run-Time Analysis

- ☐☐☐ Calculate statement execution counts and informal run-time comparison of iterative statements.

3. Class Creation

3.1 Abstraction and Program Design

- ☐☐☐ Represent the design of a program by using natural language or creating diagrams that indicate the classes in the program and the data and procedural abstractions found in each class by including all attributes and behaviors.

3.2 Impact of Program Design

- ☐☐☐ Explain the social and ethical implications of computing systems.

3.3 Anatomy of a Class

- ☐☐☐ Develop code to designate access and visibility constraints to classes, data, constructors, and methods.

3.4 Constructors

- ☐☐☐ Develop code to declare instance variables for the attributes to be initialized in the body of the constructors of a class.

3.5 Methods: How to Write Them

- ☐☐☐ Develop code to define behaviors of an object through methods written in a class using primitive values and determine the result of calling these methods.

3.6 Methods: Passing and Returning References of an Object

- ☐☐☐ Develop code to define behaviors of an object through methods written in a class using object references and determine the result of calling these methods.

3.7 Class Variables and Methods

- ☐☐☐ Develop code to define behaviors of a class through class methods.
- ☐☐☐ Develop code to declare the class variables that belong to the class.

3.8 Scope and Access

- ☐☐☐ Explain where variables can be used in the code.

3.9 this Keyword this

- ☐☐☐ Develop code for expressions that are self-referencing and determine the result of these expressions.

4. Data Collections

4.1 Ethical and Social Issues Around Data Collection

- ☐☐☐ Explain the risks to privacy from collecting and storing personal data on computer systems.
- ☐☐☐ Explain the importance of recognizing data quality and potential issues when using a data set.
- ☐☐☐ Identify an appropriate data set to use in order to solve a problem or answer a specific question.

4.2 Introduction to Using Data Sets

- ☐☐☐ Represent patterns and algorithms that involve data sets found in everyday life using written language or diagrams.

4.3 Array Creation and Access

- ☐☐☐ Develop code used to represent collections of related data using one-dimensional (1D) array objects.

4.4 Array Traversals

- ☐☐☐ Develop code used to traverse the elements in a 1D array and determine the result of these traversals.

4.5 Implementing Array Algorithms

- ☐☐☐ Develop code for standard and original algorithms for a particular context or specification that involves arrays and determine the result of these algorithms.

4.6 Using Text Files

☐☐☐ Develop code to read data from a text file.

4.7 Wrapper Classes

☐☐☐ Develop code to use ‘Integer’ and ‘Double’ objects from their primitive counterparts and determine the result of using these objects.

4.8 ArrayList Methods ArrayList

☐☐☐ Develop code for collections of related objects using ‘ArrayList’ objects and determine the result of calling methods on these objects.

4.9 ArrayList Traversals ArrayList

☐☐☐ Develop code used to traverse the elements of an ‘ArrayList’ and determine the results of these traversals.

4.10 Implementing ArrayList Algorithms ArrayList

☐☐☐ Develop code for standard and original algorithms for a particular context or specification that involve ‘ArrayList’ objects and determine the result of these algorithms.

4.11 2D Array Creation and Access

☐☐☐ Develop code used to represent collections of related data using two-dimensional (2D) array objects.

4.12 2D Array Traversals

☐☐☐ Develop code used to traverse the elements in a 2D array and determine the result of these traversals.

4.13 Implementing 2D Array Algorithms

☐☐☐ Develop code for standard and original algorithms for a particular context or specification that involves 2D arrays and determine the result of these algorithms.

4.14 Searching Algorithms

☐☐☐ Develop code used for linear search algorithms to search for specific information in a collection and determine the results of executing a search.

4.15 Sorting Algorithms

☐☐☐ Determine the result of executing each step of sorting algorithms to sort the elements of a collection.

4.16 Recursion

☐☐☐ Determine the result of calling recursive methods.

4.17 Recursive Searching and Sorting

☐☐☐ Determine the result of executing recursive algorithms that use strings or collections.

☐☐☐ Determine the result of each iteration of a binary search algorithm used to search for information in a collection.

☐☐☐ Determine the result of each iteration of the merge sort algorithm when used to sort a collection.