

SAMPLE EXAM QUESTIONS

1. Multiple Choice. Write the letter of the **best answer** in the blank to the left.

- _____ A certain Java/CN1 class named `"Point"` has constructors `"Point()"` and `"Point(int x, int y)"`. This is an example of
A. abstraction B. encapsulation C. inheritance D. overloading E. overriding
- _____ A certain Java/CN1 class named `Sphere` contains a method named `getColor()` which returns the color of the `Sphere` object. This method is an example of a (an)
A. accessor B. mutator C. aggregation D. design pattern E. abstraction
- _____ A certain Java/CN1 class named "B" extends another class named "A". Class B defines a method named "C" with the same signature as that of a method named "C" in Class A. Method C in Class B does not contain the keyword "super". A program constructs an instance of B and invokes method "C" in that object. The code which will be executed as a result of this invocation is
A. the code in A.C
B. the code in B.C
C. the code in A.C followed by the code in B.C
D. the code in B.C followed by the code in A.C
E. it depends on the code in A.C
F. it depends on the code in B.C
G. None of the above
- _____ If a Java/CN1 program contains a declaration such as `"class A {...}"`, where `"..."` represents the code defining the class, then
A. A has no parent class
B. A is its own parent
C. A is a superclass of Object
D. A is a subclass of Object
E. A is an abstraction of Object

- _____ In Java/CN1, *inheritance* is indicated using the keyword
- A. abstract
 - B. extends
 - C. implements
 - D. static
 - E. new
 - F. none of the above
- _____ Before Java 8, an *interface* consists of
- A. a set of method declarations (abstract methods)
 - B. a set of method definitions (implementations)
 - C. a class description given in an online Application Programming Interface (API)
 - D. the set of classes in an inheritance hierarchy
 - E. a set of accessor (selector and/or mutator) methods
- _____ In a UML Class Diagram depicting classes named “Student” and “Course”, a label named “takes” on the diagram would most likely represent
- A. a method in Student
 - B. a method in Course
 - C. an association
 - D. a multiplicity
 - E. a composition
- _____ In CN1, when one object is registered as containing the method(s) to be invoked when another object generates an “ActionEvent”, we say the first object is a (an)
- A. event generator
 - B. action performer
 - C. listener
 - D. layout manager
 - E. exception handler
- _____ An association between two objects named “A” and “B” such that (1) B is referenced by A but not by any other object, and (2) the lifetime of B is controlled by A, is called a (an)
- A. Composition
 - B. Aggregation
 - C. Abstraction
 - D. Encapsulation
 - E. Inheritance

_____ A CN1 build-in class *Container* is a

- A. component
- B. layout manager
- C. design pattern
- D. framework
- E. more than one of the above
- F. none of the above

[THERE WOULD BE MORE MULTI-CHOICE QUESTIONS IN THE REAL EXAM...]

2. Short Answer. Write the **best answer** in the blank to the left.

Each of the next three questions describes a class structure which uses *inheritance*. For each situation, write the word which identifies the category of inheritance usage – either *extension*, *specialization*, or *specification*. If none of these applies to a particular description, write the word *other* instead.

_____ A class *Counter* defines a method *increment(int amount)* which increments the counter value by the specified amount. A class *UnitCounter* extends *Counter* and specifies a method *increment(int amount)* which ignores the value of *amount* and always increments the counter value by 1.

_____ A class *Vehicle* declares that every instance has a way to turn itself (but it does not define how the instances are turned). A class *Truck* extends *Vehicle* and contains a method defining the way in which the Truck is turned.

_____ A class *Clock* defines methods for keeping track of the current time. A class *AlarmClock* extends *Clock* and defines methods for enabling and disabling an alarm to go off at a certain time.

Each of the next two questions describes an implementation requirement. For each situation, write the name of a design pattern which could be used to provide an appropriate solution.

_____ A program should not be allowed to create multiple *AudioPlayer* objects.

_____ A program uses a *Vector* to store a collection of objects, but the programmer wants to make sure that any subsequent decision to replace the use of *Vector* with some other data structure will not break any existing clients which use the collection.

[THERE WOULD BE MORE SHORT-ANSWER QUESTIONS IN THE REAL EXAM...]