CSUS
COLLEGE OF ENGINEERING AND COMPUTER SCIENCE
Department of Computer Science

CSc 133 — Object-Oriented Computer Graphics
John Clevenger

Name _____
(5 points)

SAMPLE MIDTERM EXAM

1. Write your name in the space above.

2. The exam is closed book, closed notes. There are 100 total points; you have 75 minutes to work on it — budget your time accordingly.

3. Absolutely NO use of ANY electronic devices is allowed during the exam. This includes cell phones, pagers, or any other communications device.

4. Please be neat — I cannot give credit to answers I cannot read.

5. The exam has 8 pages, counting this cover page. Make sure you have all the pages.

Problem	Points	Possible
1	_____	45
2	_____	10
3	_____	10
4	_____	10
5	_____	20
Total	_____	100

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

_____ A certain 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 class named **Ball** contains a method named *getColor()* which returns the color of the Ball object. This method is an example of a (an)

A. accessor B. mutator C. aggregation D. design pattern E. abstraction

_____ A certain Java 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 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 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, *inheritance* is indicated using the keyword

A. abstract
B. extends
C. implements
D. static
E. new
F. none of the above

_____ An *interface* in Java consists of

A. a set of method prototypes (signatures)
B. a set of method implementations (code bodies)
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

_____ The Java AWT is an example of a (an)

A. Design Pattern
B. Application Framework
C. Object-Oriented language
D. Abstract Data Type
E. UML package

_____ 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

_____ 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. action listener
D. layout manager
E. exception handler

_____ If you want to create an effect like a rubber-band line, which changes continually with user input, you will most likely need a (an)

A. MouseListener
B. ActionListener
C. MouseAdapter
D. MouseMotionListener
E. KeyListener

_____ Suppose a programmer creates a Java class named MyFrame which extends JFrame. Class MyFrame contains a single constructor whose body is empty. If an object of type MyFrame is created and made visible (but no other methods in the object are invoked), then when the user clicks on the "X" in the upper right corner of the window (frame),

A. the program will terminate
B. the window will be hidden but will still exist
C. the window will be closed
D. the window will be minimized ("iconified")
E. the question cannot be answered because such a class cannot be compiled
F. the answer cannot be determined from the information given

_____ 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 Swing *JPanel* is an example of a

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

_____ The advantage of extending the Swing *JPanel* class is that it provides the ability to

A. change the default layout manager
B. override paintComponent()
C. control the objects which are registered as listeners
D. add components to the panel
E. force a repaint()

2. List the responsibilities which must be implemented by each of the participants in an *observer/observable design pattern.* Identify which responsibility belongs to each type of participant.

3. Two different programming teams have implemented a class named **Rectangle**. One team provided accessors to get and set the location (origin), width, and height of a rectangle, while the other team chose to make the origin, width, and height fields public so that they can simply be directly accessed (read and/or changed). The second team argues that if you have accessors which allow you to both get and set all the values in the rectangle, there is no difference in having the fields public. Explain why the second team does not know what they are talking about. Be specific; give an example of how their approach can produce a software system that fails.

4. An interactive program displays a main frame which contains two components: a control panel containing a single button, and a separate display panel. The button has an action listener attached to it which is an instance of a separate class. The display panel has a mouse listener attached to it which is likewise an instance of a (different) separate class. The main frame is a subclass of JFrame; the panels are subclasses of JPanel, and the button is a JButton.

Draw a UML diagram depicting the associations between the elements of this program.

5. Using Java-like code, describe on the next page the structure of a program which implements a simple Graphical User Interface (GUI) consisting of the following components:

(1) a single button, whose label is initially “Hello” and which prints the message “World” on the console when pressed;

(2) a single menu with a single item which when selected has the effect of changing the button label to “Goodbye”;

(3) a single “drawing area” on which pressing a mouse button causes the background to become red, and releasing the mouse button causes the background to become blue.

You may assume that buttons have a method *setText(String)* to change their label text, and that drawing panels have a method *setBackground(Color)* to change their color.

Note: you are not being asked for a complete syntactically-correct Java program, nor for memorized details of any specific Java GUI components. Your answer should contain Java-like code fragments which describe each of the basic steps, including both initialization and control flow, which would have to appear in a program intended to operate as described above. You may choose the class organization of the program and make liberal assumptions about the names of various methods in objects and what they do, as long as your assumptions are clear.

< use the NEXT PAGE for your answer to this question >

< This page is provided for your answer to the previous question >