

Signature _____

Name _____

cs70f _____

Student ID _____

CSE 70
Final Exam
Fall 2009

Page 1 _____ (10 points)

Page 2 _____ (16 points)

Page 3 _____ (22 points)

Page 4 _____ (13 points)

Page 5 _____ (15 points)

Page 6 _____ (20 points)

Page 7 _____ (9 points)

Page 8 _____ (15 points)

Page 9 _____ (16 points)

Page 10 _____ (8 points)

Total _____ (144 points = 137 base points + 7 points EC [>5%])

More software projects have gone awry (failed) because of

- A. Lack of funding
- B. Lack of good software engineering tools
- C. Lack of testing
- D. Lack of manpower
- E. Lack of time

Probably the biggest drawback of the Waterfall Model is

What does it mean for a design (say, a class) to have high cohesion?

- A. Methods in a class access data in other classes to provide the glue that ties the whole program together
- B. Methods in a class work together on a focused task
- C. Methods in a class are grouped together because they are frequently used methods (object management optimization)
- D. Methods in a class are grouped together because they provide the same basic function (like I/O)

Upfront fully-defined requirements analysis is a big part of a Plan-Driven process like the Waterfall Model. One of the nice features of XP is there is no requirements analysis at all. True or False (Circle correct answer)

Which is considered a good and desirable design?

- A. High cohesion and low coupling
- B. High cohesion and high coupling
- C. Low cohesion and low coupling
- D. Low cohesion and high coupling

Extreme Programming (XP) is considered an example of

- A. Layered Model
- B. The version of Windows before Vista
- C. Waterfall Model
- D. Spiral Model
- E. Agile Model

A key method in the Strategy Pattern is a mutator method for each behavior variable. This allows us to

- A. Change behavior at compile time
- B. Change behavior in the interface
- C. Change behavior at run time
- D. Change state (behavior) in the object this variable is referencing

Objects maintain their own state via (Circle correct letter)

- A. Their superclass parts
- B. Their static variables
- C. Their instance variables
- D. Their local variables
- E. Their subclass parts

Interface defines _____ messages can be sent to an object while implementation defines _____ messages are coded. (Use the words "how" and/or "what" to answer this question.)

The 4 main tenets/features of OOP are

Inheritance with a concrete superclass is an example of pure inheritance of _____.

Inheritance with an interface is an example of pure inheritance of _____.

Inheritance with a(n) _____ superclass is an example of a mixture of the two types of inheritance.

_____ gives us an is-a relationship while _____ gives us a has-a relationship.

To realize the real power of dynamic binding

- A. A superclass reference must be a concrete class
- B. A superclass reference must be an interface
- C. A superclass reference must be an abstract class
- D. A superclass reference can be any type: interface, abstract class, or concrete class

In UML notation for access,

public members are notated with a _____

protected members are notated with a _____

private members are notated with a _____

In UML, how is an interface type identified in a class diagram? (Not talking about inheritance here.)

What does it mean for a design (say, a group of classes) to have low coupling?

- A. Multiple classes are coupled by inheritance of implementation low in the inheritance hierarchy
- B. Multiple classes share a common set of "global" variables for low overhead communication
- C. Multiple classes rely on the inner workings deep ("low") inside each other
- D. Multiple classes are not dependent on each other; instead they use a public interface to exchange parameter-less messages (or events)

What should you do before a commit?

General OO Principles

_____ what varies.

Favor _____ over _____.

Program to _____, not _____.

Strive for _____ coupled designs between objects that interact.

Classes should be open for _____ but closed for _____.

Depend on _____. Do not depend on _____ classes.

Only talk to your immediate _____.

Don't _____ us, we'll _____ you.

A class should have only one reason to _____.

In the MVC architecture, which part(s) plays the Observable? _____

In the MVC architecture, which part(s) plays the Observer? _____

Which is usually considered the better design in the Observer Pattern?

- A. Push data to the Observer
- B. Push data to the Observable
- C. Pull data from the Observer
- D. Pull data from the Observable

In the following Java GUI/Event Handling code fragment:

```
/* Some GUI control object that lays out GUI components */
    JButton button1 = new JButton( "Stop" );
    button1.addActionListener( new ResizableBall( /* args */ ) );
public class ResizableBall implements ActionListener
{
    /* Lots of other stuff associated with a ResizableBall */
    public void actionPerformed( ActionEvent evt )
    {
        /* Do something with evt */
    }
}
```

Which object is the Observable? _____

Which object is the Observer? _____

What type of testing allows the tester to see the inner workings of the code being tested?

What type of testing does not allow the tester to see the inner workings of the code being tested?

Given the following method/function:

```
int foo ( int x, int y )
{
  if ( x >= 25 )
  {
    if ( y <= 50 )
      return 1;
    else
      return -1;
  }
  return 0;
}
```

Possible x values: <25, 25, >25, Don't Care
Possible y values: <50, 50, >50, Don't Care
Possible return values: -1, 0, 1, Don't Care

Using the possible values to the right, list the values of x and y and expected return value to perform a full test coverage of this code.

<u>x</u>	<u>y</u>	<u>expected return value</u>
----------	----------	------------------------------

Suppose you have the following ant targets in a build.xml script (the bodies of the targets are not important):

```
<target name="come-to-class" />
<target name="get-enough-sleep" />
<target name="get-cup-of-coffee" />
<target name="stay-awake-in-class" depends="get-enough-sleep" />
<target name="pay-attention-in-class" depends="come-to-class,stay-awake-in-class" />
<target name="study-hard" />
<target name="get-good-grades" depends="pay-attention-in-class,study-hard" />
```

List all the targets in the order they would be run by ant if you entered the command:

```
ant get-good-grades
```

Name at 2 Bad Smells and what Refactoring techniques should be used to eliminate each.

Bad Smell #1:

Refactoring technique(s):

Bad Smell #2:

Refactoring technique(s):

Given the following Java code, draw the UML class diagrams that model the structure of the code. Do not worry about any of the details in the objects with the exception you do specify the variable ref in class Bar and its relationship with class Bang. Label any lines you draw between class diagrams with "is-a" or "has-a" as appropriate.

```
public class Bang { ... }  
  
public interface Foo { ... }  
  
public class Fubar { ... }  
  
public class Bar extends Fubar implements Foo  
{  
    private Bang ref = new Bang();  
  
    // other code ...  
}
```

Match each pattern with its description:

Pattern
___ Strategy
___ Adapter
___ Iterator
___ Facade
___ Composite
___ Observer
___ Template Method
___ Decorator
___ State
___ Proxy
___ Factory Method
___ Singleton
___ Command

- A) Wraps another object and provides a different interface to it
- B) Wraps another object and provides additional behavior for it
- C) Encapsulates state-based behavior and delegate behavior to the current state
- D) Wraps another object to control access to it
- E) Wraps a bunch of objects to simplify their interface
- F) Subclasses decide how to implement steps in an algorithm
- G) Subclasses decide which concrete classes to create
- H) Ensures one and only one object is created
- I) Clients treat collections of objects and individual objects uniformly
- J) Encapsulates interchangeable behaviors and uses delegation to decide which behavior to use
- K) Provides a way to traverse a collection of objects without exposing its implementation
- L) Encapsulates a request as an object
- M) Allows objects to be notified when state changes

Which design pattern is commonly used with the MVC architecture? _____

Which design pattern is commonly used with Java Event Handling? _____

What does MVC stand for? _____

Which design pattern allows for "hook" methods to "hook into" the design at various points? _____

Which design pattern does RMI use? _____

What does RMI stand for? _____

If you have an interface defined in your early design, why might it be beneficial to change the interface into an abstract class instead? What does using an abstract class buy you where otherwise you could just use an interface?

To make Christmas look like Thanksgiving, we need to provide an adapter interface

- A) with the Christmas methods and implement them with calls to equivalent Thanksgiving methods
- B) that throws an UnsupportedOperationException each time the client tries to use any Thanksgiving method
- C) with the Thanksgiving methods and implement them with calls to equivalent Christmas methods
- D) that throws an UnsupportedOperationException each time the client tries to use any Christmas method
- E) with both Christmas and Thanksgiving methods and implement them with calls to each other's equivalent methods
- F) that uses reflection to create a Christmas object each time the client requests a Thanksgiving object

The OO Principle: "One class, one responsibility" can also mean we want classes to have

- A) High cohesion
- B) Low cohesion
- C) Tight coupling
- D) Loose coupling

The Java enhanced-for loop uses which design pattern? _____

Which OO Principle is also known as the Hollywood Principle?

Which OO Principle is also known as the Principle of Least Knowledge (or the Law of Demeter)?

Which OO Principle is also known as the Dependency Inversion Principle?

Some AC adaptors do more than just change the interface – they add other features like surge protection, indicator lights, and other bells and whistles. If you were going to implement these kinds of features, what pattern would you use? _____

A common way to implement a history of operations to provide multiple undo operations is with

Which is often considered a better design choice?

- A) Using a null object
- B) Using the null reference

In the Singleton pattern, we declare all constructors _____ and declare the public simple factory-like methods _____ .

Which allows us to change behavior dynamically at runtime?

- A) Object composition
- B) Object inheritance

The Java I/O libraries are largely based on the _____ pattern.

What is the name of the tool that monitors a program execution and tells you where it is consuming time and space? _____

I talked about a standard Java class that incorrectly uses inheritance instead of composition. Which standard Java class is this? _____

In the Web Apps security talk, what was probably the most common security problem? _____

In Java, instance variables live in objects and all objects live in the _____. Static variables are stored in the type descriptor for that type which lives in the _____. Local variables and formal parameters live in a data structure called _____.

Interfaces can only contain public _____ methods and public static _____ constants. If a class definition contains at least one abstract method, then the class must be declared to be _____.

Which architectural style is appropriate when several independent program components must access common persistent data?

- A) Layered
- B) Pipe-and-Filter
- C) Shared-Data
- D) Event-Driven
- E) MVC

Which architectural style is an excellent model for component behavior when the program as a whole transforms a stream of input into a stream of output?

- A) Layered
- B) Pipe-and-Filter
- C) Shared-Data
- D) Event-Driven
- E) MVC

Which architectural style is a good choice for a program that must react to unpredictable sequences of input?

- A) Layered
- B) Pipe-and-Filter
- C) Shared-Data
- D) Event-Driven
- E) MVC

Which architectural style is the OSI network model an example of?

- A) Layered
- B) Pipe-and-Filter
- C) Shared-Data
- D) Event-Driven
- E) MVC

Write a simple svn command to bring the latest changes from a Subversion repository into your working copy.

Write a simple svn command to see which files you have locally modified in a working copy of a Subversion repository.

Agile development values working software over comprehensive documentation. T or F

Agile development values contract negotiation over customer collaboration. T or F

Agile development values processes and tools over individuals and interactions. T or F

Agile development values responding to change over following a plan. T or F

Polymorphism in Java is realized by

- A) Subclassing from a concrete class
- B) Subclassing from an abstract class
- C) Implementing one or more interfaces
- D) A and B only
- E) A, B, and C

Encapsulation tends to _____ coupling.
(increase / decrease)

In general, good software engineering principles dictate the use of only _____ and _____
Java access modifiers and not _____ or _____ access modifiers.

UML sequence diagrams show inheritance relationships. T or F

Polymorphism uses the type of the object at run time vs. the type of the reference at compile time to decide which implementation of a method to bind to. T or F

In Java, the 3 keywords to turn off the default dynamic run time binding and turn on static compile time binding are

The tool _____ uses a file named build.xml that specifies various _____ that can be invoked.

What is Rick's dog's name? _____

In general, creating a new object passing all the initial values in to a single constructor vs. reusing an existing object and calling multiple mutator (set) methods to reset that object's instance variables is

- A) about the same cost
- B) creating a new object is more expensive
- C) calling multiple mutator methods is more expensive

In general, a non-synchronized method vs. an equivalent synchronized method is

- A) about the same cost
- B) synchronized method is more expensive
- C) non-synchronized method is more expensive