2010 Midterm Exam

Answer all 10 questions. Answer essay questions as briefly as possible.

The following might be names of patterns: Abstract Class, Abstract Factory, Adapter, Application Controller, Bridge, Builder, Chain of Responsibility, Collaborator, Command, Composite, Decorator, Façade, Factory Method, Flyweight, Interpreter, Iterator, Master-Slave, Mediator, Memento, Null Object, Observer, Prototype, Proxy, Singleton, Specification, State, Strategy, Template Method, Visitor.

1. The purpose of many of the design patterns is to make it easy to change some property of the system. What design pattern would you use to make it easy to change:

   A. The algorithm that an object uses.

   B. The class of the object that a method returns.

   C. The kind and number of objects that react to changes in the object you are designing.

   D. Adding operations to classes without changing the class.

   E. How methods in a class behave.

2. What is a code smell? List two code smells.

3. Explain shallow copy and deep copy.

4. Explain control coupling. Give an example.

5. Explain Coupling and Cohesion.

6. According to the Big Ball of Mud paper why should we use piecemeal growth?

7. Explain the command processor pattern.

8. What is the difference between the Proxy and Decorator pattern?

9. Explain how the State pattern works.

10. What are some of the problems cause by using the singleton pattern?

2010 Final Exam

Answer all 10 questions. Answer essay questions as briefly as possible.

The following might be names of patterns: Abstract Class, Abstract Factory, Active Object Model, Adapter, Application Controller, Bridge, Builder, Chain of Responsibility, Collaborator, Command, Composite, Decorator, Dependency Injection, Dynamic Factory, Façade, Factory Method, Flyweight, Interpreter, Iterator, Master-Slave, Mediator, Memento, Null Object, Observer, Property, Prototype, Proxy, Singleton, Schema, Smart Variables, Specification, State, Strategy, Template Method, Type Object, Visitor.

1. List three patterns that can be used to extend the functionality of a class without subclassing it.

2. Circle the correct answer in the following.

True False Abstract Factory methods are often implemented with factory methods.

True False The Factory Method pattern is used when a class can't anticipate the class of objects it must create.

True False The builder pattern uses an abstract class for products it builds.

True False The Memento pattern allows the Originator to store a history of mementos.

True False The Interpreter pattern deals with parsing language expressions.

True False In the Facade pattern clients are not allowed to use subsystem classes directly.

3. The Singleton pattern insures that clients only access one instance of a class. One can also insure that clients only access one instance of a class by implementing the class using only static fields and static methods. What advantage(s) does the Singleton pattern have over using a class with all static methods and static fields?

4. This question is about the Property pattern from the Metadata and Active Object Models paper.
   a. What is the purpose (intent) of the Property pattern?

    b. How does one implement the pattern in Java?
    c.

5. Give two situations where the Type Object Pattern would be applicable. (That is when is the Type Object Pattern applicable)

6. Design patterns have consequences, some good and some bad.
    a. Give one good consequence of the Chain of Responsibility pattern.

    b. Give one bad (or negative) consequence of the Chain of Responsibility pattern.

7. The text claims that Mediator and Observer are competing patterns.
    a. Explain how they compete with each other.

    b. What are the strengths of each pattern over the other.

8.
    a. What is the difference between intrinsic and extrinsic state?

    b. Give an example of each.

9.
    a. What is the difference between an object adapter and a class adapter?

    b. What is an advantage of a class adapter over an object adapter?

10. In the Composite pattern one can place the child management operations (add, remove) either in the Composite class or the Component class. What are the trade-offs of putting them in the Component class?

11. Explain how the Bridge pattern works.

2011 Midterm

Answer all 10 questions. Answer essay questions as briefly as possible.

The following might be names of patterns: Abstract Class, Abstract Factory, Adapter, Application Controller, Bridge, Builder, Chain of Responsibility, Collaborator, Command, Composite, Decorator, Façade, Factory Method, Flyweight, Interpreter, Iterator, Master-Slave, Mediator, Memento, Null Object, Observer, Prototype, Proxy, Singleton, Specification, State, Strategy, Template Method, Visitor.

1. The purpose of many of the design patterns is to make it easy to change some property of the system. What design pattern would you use to make it easy to change:

   A. The algorithm that an object uses.

   B. The class of the object that a method returns.

   C. The kind and number of objects that react to changes in the object you are designing.

   D. Adding operations to classes without changing the class.

   E. How methods in a class behave.

2. What is a code smell? List two code smells.
3. In the Big Ball of Mud paper what do they claim are the reasons to use Piecemeal Growth? What is the biggest danger of using Piecemeal Growth?

4. What are the advantages and disadvantages of the Null Object pattern.

5. Explain Coupling and Cohesion.

6. Explain.

   A. Information Hiding

   B. Encapsulation

   C. Abstraction

7. Explain one of the following types of coupling: Data Coupling, Control Coupling, Inside Internal Object Coupling.

8. Select either the Visitor pattern or the Decorator pattern and explain how the pattern works.

9. The design patterns text claims it uses two main design principles. Select one pattern that uses both principles. Show how it uses both.

10. What are the advantages and disadvantages of the Command pattern.

2011 Final

Answer all 18 questions. Answer essay questions as briefly as possible.

1. (10 points) Circle the correct answer for each of the following.

True False The facade pattern is used to lower coupling between different subsystems.

True False In the facade pattern clients only interact with a subsystems facade. They never interact with the subsystem directly.

True False The facade object only forwards a client request to the subsystem, it does not answer the request itself.

True False The flyweight pattern is used to save storage when you have large sized objects.

True False The flyweight pattern can only be used when the application does not depend on object identity.

True False The text states that one uses the mediator when reusing an object is difficult because it refers to and communicates with many other objects.

True False The mediator pattern requires the abstract Mediator class.

True False The mediator centralizes control.

True False It's often best to implement State objects as flyweights.

True False The builder pattern requires an abstract class or interface for the product.

2. (10 points) Circle the correct answer for each of the following.

True False In the builder pattern the director selects the concrete builder.

True False The builder pattern lets you vary a product's internal representation.

True False In the chain of responsibility more than one handler can handle a particular request.

True False In the chain of responsibility a request will be handled.

True False Chain of responsibility should be use when who handles a request should be specified dynamically.

True False The interpreter pattern can be used for complex grammars.

True False In the interpreter pattern is it easy to extend grammar.

True False The interpreter pattern is not efficient.

True False A prototype manager is required in the prototype pattern.

True False The prototype pattern can reduce subclassing.

3. (10 points) Circle the correct answer for each of the following.

True False In the proxy pattern the proxy does not create the real subject.

True False The singleton pattern is an improvement over global variables.

True False The singleton pattern permits a variable number of instances.

True False The abstract factory pattern promotes consistency among products.

True False Supporting new kinds of products in difficult when using the abstract factory pattern.

True False In implementing an abstract factory the singleton pattern is often used.

True False Mementos might be expensive.

True False The Memento pattern allows the Originator to store a history of mementos.

True False The composite pattern can make your design overly general.

True False The composite pattern makes the client simple.

The following might be names of patterns: Abstract Class, Abstract Factory, Active Object Model, Adapter, Application Controller, Bridge, Builder, Chain of Responsibility, Collaborator, Command, Composite, Decorator, Dependency Injection, Dynamic Factory, Façade, Factory Method, Flyweight, Interpreter, Iterator, Master-Slave, Mediator, Memento, Null Object, Observer, Property, Prototype, Proxy, Singleton, Schema, Smart Variables, Specification, State, Strategy, Template Method, Type Object, Visitor.

4. (2 points) What design pattern would you use to make it easy to change the implementation of an abstraction?

5. (2 points) What design pattern should you think of when you want to hide how you construct a complex object?

6. (2 points) What design pattern should you think of when when you wish to send a method to one of several objects without specifying the receiver explicitly?

7. (2 points) What design pattern would you use when you have a group of related objects that are designed to work together and you need to insure that they are used together?

8. (2 points) Which design pattern would you use when you want a client to create a new object without explicitly specifying the class of the new object?

9. (2 points) What design pattern would you use to make it easy to change the algorithm a class uses?

10. (2 points) What design pattern would you use when more that one object may handle a request and you don't know in advance which object will handle a particular request?

11. (2 points) What design pattern might you use when you wish to reduce tight coupling between classes?

12. (2 points) What design pattern would you use when you only have the binary of a class and need to modify the signature of some of the methods in the class?

13. (2 points) What design pattern would you use to reduce dependence on hardware and/or software platforms?

14. (10 points) The text claims that Mediator and Observer are competing patterns.

    a. Explain how they compete with each other.

    b. What are the strengths of each pattern over the other.

15. (10 points)
    a. What is the difference between intrinsic and extrinsic state?

    b. Give an example of each.

16. (10 points) Design patterns have consequences, some good and some bad.
    a. Give one good consequence of the Singleton pattern.

    b. Give one bad (or negative) consequence of the Singleton pattern.

17. (10 points) In object-oriented programming one is advised to avoid case (and if) statements. Select one design pattern that helps avoid case statements and explain how it helps.

18. (10 points) Decorator and Chain of Responsibility are similar because they both contain multiple nested classes. In other respects, they are quite different. Explain the differences between Decorator and Chain of Responsibility.