

**CIS 350 – Introduction to Software Engineering**  
**Topic: OOAD, Design Patterns, MVC & MVP**  
**Winter 2013, March 31, 2013**

**Due: Wednesday, April 3, 2013**

**Student Name:** \_\_\_\_\_

1. \_\_\_\_\_ classes model the interaction between the system's surroundings and its inner workings.  
a) Control                      b) Boundary                      c) Entity
2. \_\_\_\_\_ classes store and manage information in a system.  
a) Control                      b) Boundary                      c) Entity
3. \_\_\_\_\_ classes coordinate the behavior of a use case.  
a) Control                      b) Entity                      c) Boundary
4. \_\_\_\_\_ classes contain business/domain logic and may deal with persistent storage.  
a) Entity                      b) Control                      c) Boundary
5. \_\_\_\_\_ class delegates the work of a use case to other classes.  
a) Entity                      b) Control                      c) Boundary
6. \_\_\_\_\_ diagrams are used to realize a specific scenario of a use case.  
a) Class                      b) Sequence                      c) Deployment                      d) Component
7. Noun extraction is a technique for identifying \_\_\_\_\_ classes.  
a) Entity                      b) Control                      c) Boundary
8. \_\_\_\_\_ technique uses index cards to identify the responsibilities and collaborators of classes in a system.  
a) Noun extraction                      b) CRC                      c) Verb extraction
9. The participants in a CRC session walkthrough one or more \_\_\_\_\_ to stress test design of classes in a system.  
a) scenarios                      b) streets                      c) information                      d) data
10. \_\_\_\_\_ refers to the bundling of data with the operations that operate on that data.  
a) Information hiding                      b) Encapsulation                      c) Abstraction                      d) Stepwise refinement

11. Module \_\_\_\_\_ is a measure of interdependence between two modules.  
a) cohesion                      b) coupling
12. High coupling and low cohesion are desirable qualities of a good design.  
a) True                              b) False
13. Coincidental cohesion for a module is very desirable.  
a) True                              b) False
14. Composition is a method of reuse that leads to “white-box” reuse.  
a) True                              b) False
15. Two modules are \_\_\_\_\_ coupled if one module directly references or changes the contents of another module.  
a) data                              b) content                      c) common                      d) control
16. What GoF category of patterns does Singleton pattern belong to?  
a) Structural                      b) Creational                      c) Behavioral
17. \_\_\_\_\_ design patterns abstract the object instantiation process.  
a) Structural                      b) Creational                      c) Behavioral
18. \_\_\_\_\_ pattern enables us to use a complex system more easily, either to use just a subset of the system or use the system in a particular way.  
a) Singleton                      b) Facade                      c) Observer                      d) Proxy
19. Observer pattern is an example of \_\_\_\_\_ pattern.  
a) Structural                      b) Creational                      c) Behavioral
20. Java supports the implementation of Observer pattern using  
a) Dependent &                      b) Observable class &                      c) Publish &                      d) WatchMe &  
Dependee classes                      Observer interface                      Subscriber classes                      WatchingYou classes
21. \_\_\_\_\_ pattern is used when you need to notify a varying list of objects that an event has occurred.  
a) Facade                              b) Singleton                      c) Observer                      d) Proxy
22. In Java’s implementation of the Observer pattern, the observers provide a common interface to the subject (observable) by implementing the \_\_\_\_\_ method.  
a) notifyObservers                      b) update                      c) addObserver                      d) deleteObserver

23. In MVC triad, the \_\_\_\_\_ is responsible for managing the application data or state.
- a) view                      b) model                      c) controller                      d) presenter
24. In MVC triad, the \_\_\_\_\_ is responsible for mapping user actions or gestures into operations on the model.
- a) view                      b) presenter                      c) controller
25. In MVC triad, the \_\_\_\_\_ is responsible for rendering the model's content and sending user action to the controller.
- a) view                      b) presenter                      c) controller                      d) model
26. In \_\_\_\_\_ architecture, the view is strictly passive, thin, and shallow.
- a) MVC                      b) MVP
27. In \_\_\_\_\_ architecture, the view and model are fully decoupled and are unaware of each other.
- a) MVC                      b) MVP