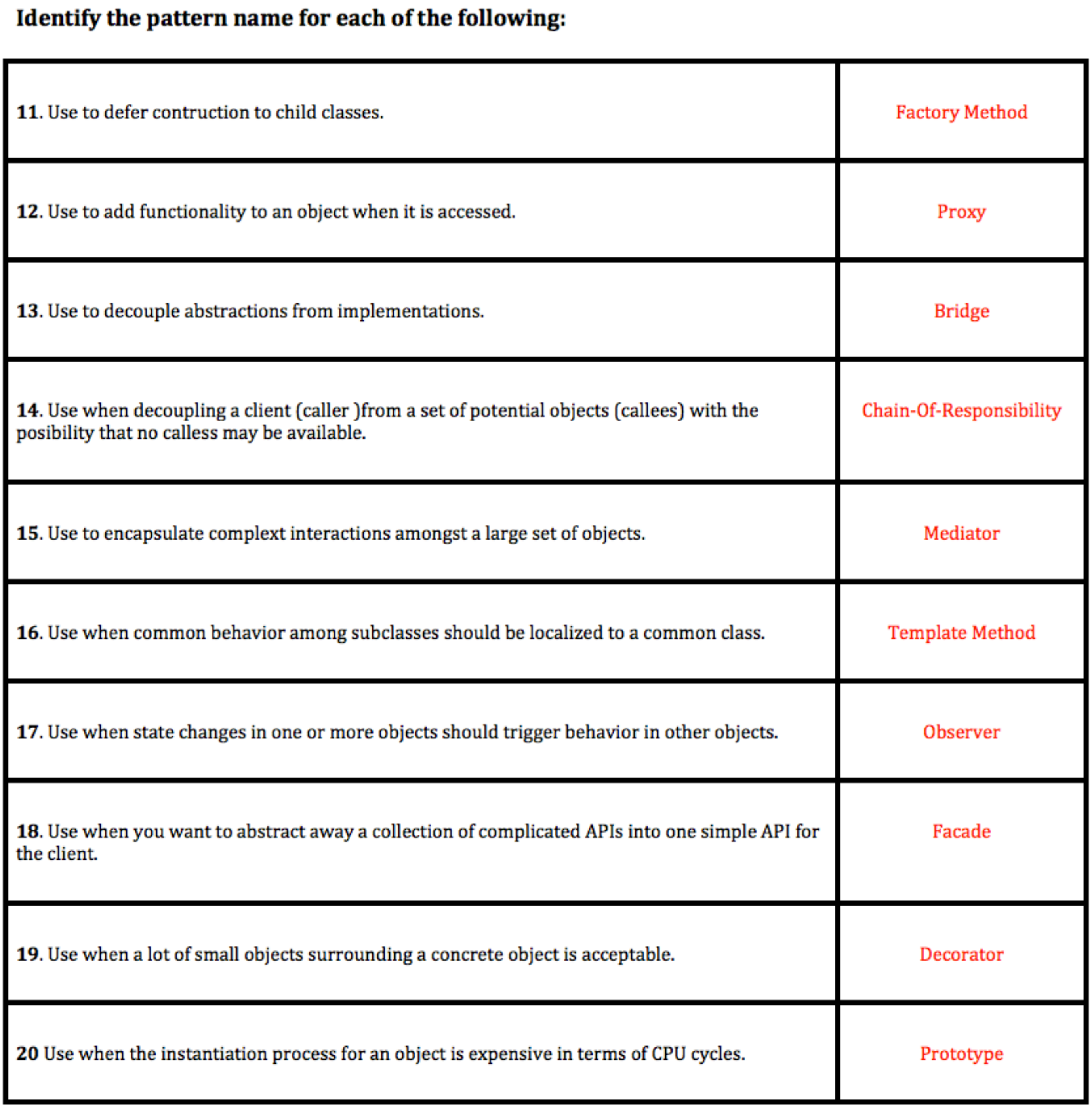
1.1: Identify the pattern name for each of the following:

10 pts



Identify the pattern name for each of the following:

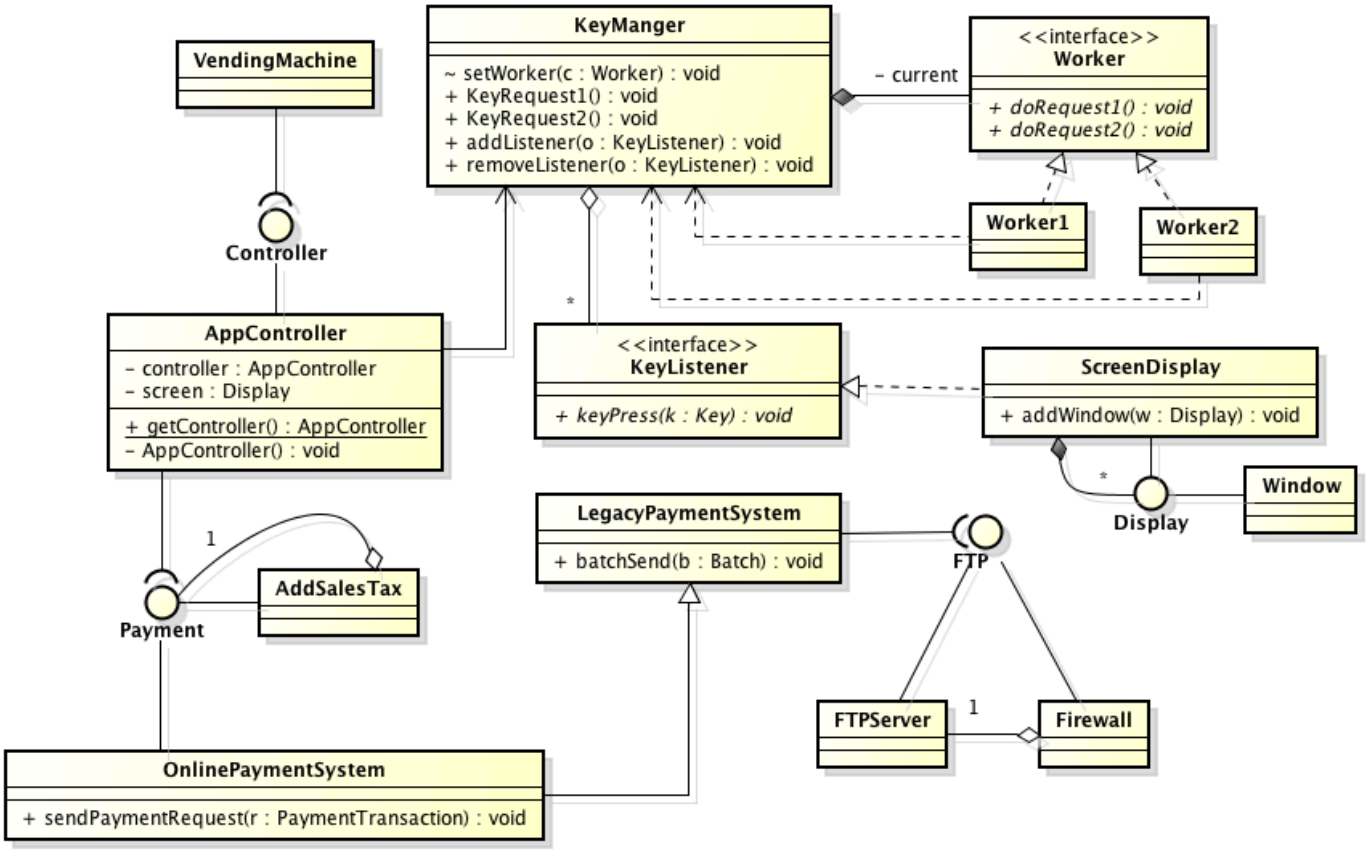
11. Use to defer contruction to child classes.	Factory Method
12. Use to add functionality to an object when it is accessed.	Proxy
13. Use to decouple abstractions from implementations.	Bridge
14. Use when decoupling a client (caller)from a set of potential objects (callees) with the posibility that no calless may be available.	Chain-Of-Responsibility
15. Use to encapsulate complext interactions amongst a large set of objects.	Mediator
16. Use when common behavior among subclasses should be localized to a common class.	Template Method
17. Use when state changes in one or more objects should trigger behavior in other objects.	Observer
18. Use when you want to abstract away a collection of complicated APIs into one simple API for the client.	Facade
19. Use when a lot of small objects surrounding a concrete object is acceptable.	Decorator
20 Use when the instantiation process for an object is expensive in terms of CPU cycles.	Prototype

For Questions 6 to 15, give the name of the pattern that would be used to solve the problems described.

Question #	Problem Description	Pattern Name
6	Want to be able to add functionality at run-time to objects but not add the behavior to the class.	Decorator
7	A way for two collaborating object hierarchies to evolve independently.	Abstract Factory
8	Need for a way to decouple the sender and receivers of a message.	CoR or Command
9	When parents classes must let child classes create objects.	Factory Method
10	This can be used to make a complex API simpler and more developer friendly.	Facade
11	Want to add tracing and audit functionality to objects whenever they are accessed without changing the object's behavior.	Proxy
12	When a vendor's class must be modified, but cannot, in order to integrate with new functionality.	Adapter
13	When a publish/subscriber model amongst objects is needed.	Observer
14	An object would like to process a request, but it is not currently empowered with the data needed to perform the responsibility.	Command
15	Would like for a way to make it seemed as though an object changes its class at runtime.	State

Answer for Question #1.3

25 pts



Patterns:

- Observer: KeyManager (Subject), ScreenDisplay (Observer)
- Decorator: OnlinePaymentSystem (Base Component), AddSalesTax (Decorator)
- Adapter (Class): LegacyPaymentSystem (Parent Adaptee), OnlinePaymentSystem (Adapter)
- Composite: ScreenDisplay (Composite), Window (Leaf), AppController (Client)

[Extra]

- State Pattern: KeyManager (Context/State Machine), Worker1, Worker2 (States), Worker (State Interface)
- Singleton: AppController
- Proxy: FTPServer (Target Subject), Firewall (Proxy), LegacyPaymentSystem (Client)