

## **Design Patterns**

1. Decorator pattern will be best suited for this problem as it allows us to apply number of toppings one on top of other. And as such the topping classes can be implemented as decorator classes.
2. Observer pattern will be best suited for this problem as it involves notifying classes as soon as any change in the subject occurs.
3. Singleton pattern will be best suited for this problem because we have a requirement that only one instance of the class should be created and all the threads should be working on the same value.
4. Strategy pattern will be best suited here because we have multiple algorithms for a specific task and the client decides the actual implementation to be used at the runtime. And the algorithms vary independently from the clients that wish to use it.