**Name:** Strategy

**Problem:**
In some situations you may have different approaches for implementing the same method that you want to use based on the situation.

In other words, your program needs to respond differently to a method under different situations.

Can be used in a situation where you wish to change the algorithms employed at different times.

**Solution:**

![Diagram](image)

**Consequences:**
Enhances encapsulation, by implementing each algorithm in its own class.

Completely overhauling your program is easier, simply employ a new Strategy.

Allows for clearly understandable code by modeling a family of approaches that can all be used to solve the same problem.

Simplifies maintenance as new strategies can be easily added by creating a new subclass and does not affect existing code.