* Sonya Yuen- Controller Pattern (GRASP)

Text

Description automatically generated

* + The above code follows GRASP’s controller principle as the “GameMapController” class represents the overall system of the game and also delegates work to other classes such as the tower and enemy classes. Having this controller class is benefitting out design by providing the opportunity to have reuse potential and improved project extension.
* Evan Leleux - Liskov Substitution Principle (SOLID)

Text

Description automatically generated

Text

Description automatically generated

* + The Tower class and the resulting children follow the Liskov Substition Principle of SOLID because the Tower class was extended for each unique tower and each unique tower implemented every element of the parent Tower class so they can be used interchangeably with Polymorphism.
* Andrew Fitton – Single Responsibility Principle (SOLID principle):

Text

Description automatically generated

This is Single Responsibility principle because this (and the other tower classes) only has one responsibility within the program. This class is meant to create establish the Boston Eagle enemies, and nothing else and thus it follows the Single Responsibility principle.

* Erich Drawdy – Open Closed Principle (SOLID principle):

Text

Description automatically generatedText

Description automatically generated

This is the open closed principle because we utilized polymorphism via abstract classes and extending (from generic towers to specific towers) in order to be open to extension buy not modification. Behaviors utilized by all classes are put into the tower class, then the electric tower (and other classes) extend to this and implement them in their own ways.

* Braden Dunaway – Creator Principle (GRASP)

Text

Description automatically generatedText

Description automatically generated

This is a good example of how we implemented the Creator principle from GRASP in our project. The EnemyHandler class aggregates the enemies in the game and as such is responsible for creating the instances of said enemies in it’s constructor definition along with it’s getNewEnemy() method.