

SU Opgave 6I

Artur Barcij [hvx912]

March 18, 2022

1

Single responsibility principle is made for one change only. This isolates it from other changes. For example in Galaga is run game loop made for only change. To run game or to stop it.

2

Open-closed principle is made to change code without recompiling it. This could be for example animations or shooting in Galaga, where we see enemy to explode, thus their code changing without recompiling. Or when player shoots, the bullets also are seen or unseen, this changes code too.

3

Liskov substitution principle is about changing superclass with subclass.

4

Interface segregation principle is about holding a lot of code private, thus making interface user-friendly. ISP means that interfaces should not be dependable on methods, which are not applied to them, thus making a lot of big interfaces into smaller ones.

5

Dependency inversion principle is focused on inversion. DIP states that code should not be dependable on each other, but on abstraction. Abstraction should not be dependable on details, thus concrete implementation.