Design and Implementation of a Class

Review of chapter 1-4
Design

- Analyze the problem and identify the classes in the problem
- Define the classes discovered for solving the problem
- Each class is defined by a class diagram
- Class diagram has
  - Class name
  - Data/characteristics/fields/has a ...
  - Methods or operations/behaviors/can do...
  - Follow the naming conventions
Implementation

• Import necessary APIs
• Begin the Class definition
• Define data
• Define methods
Let's review the Crab class

import Greenfoot.*; // import the function of Greenfoot API

Class Crab extends Animal
{
    // data: mostly private (encapsulation)
    private int numWormsEaten;
    private GreenfootImage image1;
    private GreenfootImage image2;
    // any other data
Methods

• methods: can be public, private, or protected
• anything to be accessible from outside is public
• any method/function for local use is private
• any methods usable by subclasses not by public is protected
• methods have return value (void, int etc.)
• methods have parameters (..)
• method scope is indicated by {  }
Lets go through the Crab Methods

```java
public void act()
{
...
}

private void checkKeyPress()
{

}

private void eatWorms()
{

}
```
Inheritance Hierarchy

- World
  - CrabWorld
- Actor
  - Animal
    - Crab
    - Lobster
    - Worm
Now let's implement the complete code for Crab class

• ....

• We demonstrated the operation of this code in the last few lectures.

• This lecture brings it all foundational concepts together before we move on newer concepts.