

LAPORAN HASIL PRAKTIKUM
JOBSHEET 9 POLIMPORPHISM

PEMROGRAMAN BERBASIS OBJEK



Disusun Oleh :

RASENDRIYA DAFA SETIADI

NIM. 2341720125

TI-2E / 25

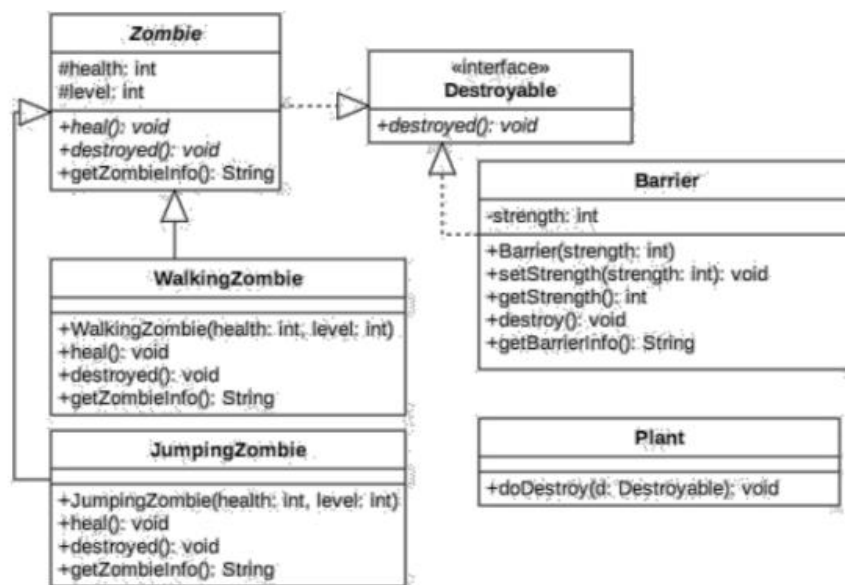
D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

2024/2025

Dalam suatu permainan, Zombie dan Barrier bisa dihancurkan oleh Plant dan bisa menyembuhkan diri. Terdapat dua jenis Zombie, yaitu Walking Zombie dan Jumping Zombie. Kedua Zombie tersebut memiliki cara penyembuhan yang berbeda, demikian juga cara penghancurannya, yaitu ditentukan oleh aturan berikut ini:

- Pada WalkingZombie
 - Penyembuhan : Penyembuhan ditentukan berdasar level zombie yang bersangkutan
 - Jika zombie level 1, maka setiap kali penyembuhan, health akan bertambah 10%
 - Jika zombie level 2, maka setiap kali penyembuhan, health akan bertambah 30%
 - Jika zombie level 3, maka setiap kali penyembuhan, health akan bertambah 40%
 - Penghancuran : setiap kali penghancuran, health akan berkurang 2%
- Pada Jumping Zombie
 - Penyembuhan : Penyembuhan ditentukan berdasar level zombie yang bersangkutan
 - Jika zombie level 1, maka setiap kali penyembuhan, health akan bertambah 30%
 - Jika zombie level 2, maka setiap kali penyembuhan, health akan bertambah 40%
 - Jika zombie level 3, maka setiap kali penyembuhan, health akan bertambah 50%
 - Penghancuran : setiap kali penghancuran, health akan berkurang 1%

Buat program dari class diagram di bawah ini!



- Destroyable.java

```
public interface Destroyable {  
    void destroyed();  
}
```

- Barrier.java

```
public class Barrier implements Destroyable {  
  
    private int strength;  
  
    public Barrier(int strength) {  
        this.strength = strength;  
    }  
    public int getStrength() {  
        return strength;  
    }  
    public void setStrength(int strength) {  
        this.strength = strength;  
    }  
    public void destroyed() {  
        strength -= strength*0.1;  
    }  
    public String getBarrierInfo() {  
        return "\nBarrier strength= " + strength;  
    }  
}
```

- Zombie.java

```
abstract class Zombie implements Destroyable {  
  
    protected int health;  
    protected int level;  
  
    public Zombie(int health, int level) {  
        this.health = health;  
        this.level = level;  
    }  
  
    public abstract void destroyed();  
    public abstract void heal();  
    public String getZombieInfo() {  
        return "Health = " + this.health + "\n"+  
            "Level = " + this.level;  
    }  
}
```

- WalkingZombie.java

```
public class WalkingZombie extends Zombie{
    WalkingZombie(int health,int level){
        super(health, level);
    }
    @Override
    public void heal(){
        System.out.println("WalkingZombie healing...");
        if (this.level == 1) {
            this.health += health * 0.20;
        } else if (this.level == 2) {
            this.health += health * 0.30;
        } else if (this.level == 3) {
            this.health += health * 0.40;
        }
    }
    @Override
    public String getZombieInfo(){
        return "Walking Zombie Data ="+"\n"+" Health: " + this.health + "\n"+"Level: "
    }
    @Override
    public void destroyed(){
        health -= health * 0.19;
    }
}
```

- JumpingZombie.java

```
public class JumpingZombie extends Zombie{
    JumpingZombie(int health,int level){
        super(health,level);
    }

    @Override
    public void destroyed() {
        health -= health * 0.091;
    }
    public void heal(){
        System.out.println("Jumping Zombie healed!");
        if (this.level == 1) {
            this.health += health * 0.30;
        } else if (this.level == 2) {
            this.health += health * 0.40;
        } else if (this.level == 3) {
            this.health += health * 0.50;
        }
    }
    @Override
    public String getZombieInfo(){
        return "Jumping Zombie Data ="+"\n"+" Health: " + this.health + "\n"+"Level: "
    }
}
```

- Plant.java

```
public class Plant {  
    public void doDestroy(Destroyable d) {  
        if(d instanceof WalkingZombie)  
            d.destroyed();  
        else if(d instanceof JumpingZombie)  
            d.destroyed();  
        else if(d instanceof Barrier)  
            d.destroyed();  
    }  
}
```

- Tester.java

```
public static void main(String[] args) {  
    WalkingZombie wz = new WalkingZombie(100, 1);  
    JumpingZombie jz = new JumpingZombie(100, 2);  
    Barrier b = new Barrier(100);  
    Plant p = new Plant();  
    System.out.println(""+wz.getZombieInfo());  
    System.out.println(""+jz.getZombieInfo());  
    System.out.println(""+b.getBarrierInfo());  
    System.out.println("-----");  
    for(int i=0;i<4;i++){  
        p.doDestroy(wz);  
        p.doDestroy(jz);  
        p.doDestroy(b);  
        System.out.println(""+wz.getZombieInfo());  
        System.out.println(""+jz.getZombieInfo());  
        System.out.println(""+b.getBarrierInfo());  
    }  
}
```

Output

```
Barrier strength= 81
Walking Zombie Data =
  Health: 52
Level: 1
Jumping Zombie Data =
  Health: 73
Level: 2
```

```
Barrier strength= 72
Walking Zombie Data =
  Health: 42
Level: 1
Jumping Zombie Data =
  Health: 66
Level: 2
```

```
Barrier strength= 64
```