Университет ИТМО

Мегафакультет компьютерных

технологий и управления

Факультет программной инженерии и

компьютерной техники

**Лабораторная работа №2**

По программированию

Вариант: 92775

Выполнил: Зыонг Динь Ань

Группа: P3107

Преподователь: Гаврилов А.В.

Санкт-Петербург

2022

Оглавление

[Задание: 3](#_Toc118289233)

[Покемоны: 3](#_Toc118289234)

[Диаграмма классов: 4](#_Toc118289235)

[Кодпрограммы: 5](#_Toc118289236)

[Battleground.java 5](#_Toc118289237)

[Коды покемон: 6](#_Toc118289238)

[Genesect.java 6](#_Toc118289239)

[Deerling.java 6](#_Toc118289240)

[Sawbuck.java 6](#_Toc118289241)

[Oddish.java 7](#_Toc118289242)

[Gloom.java 7](#_Toc118289243)

[Vileplume.java 7](#_Toc118289244)

[Коды move: 8](#_Toc118289245)

[DoubleTeam.java 8](#_Toc118289246)

[EnergyBall.java 8](#_Toc118289247)

[MetalClaw.java 9](#_Toc118289248)

[MetalSound.java 9](#_Toc118289249)

[Rest.java 10](#_Toc118289250)

[Screech.java 11](#_Toc118289251)

[SleepPowder.java 11](#_Toc118289252)

[Swagger.java 12](#_Toc118289253)

[SwordsDance.java 12](#_Toc118289254)

[Thunderbolt.java 13](#_Toc118289255)

[Результат работы: 13](#_Toc118289256)

[Вывод: 17](#_Toc118289257)

# Задание:

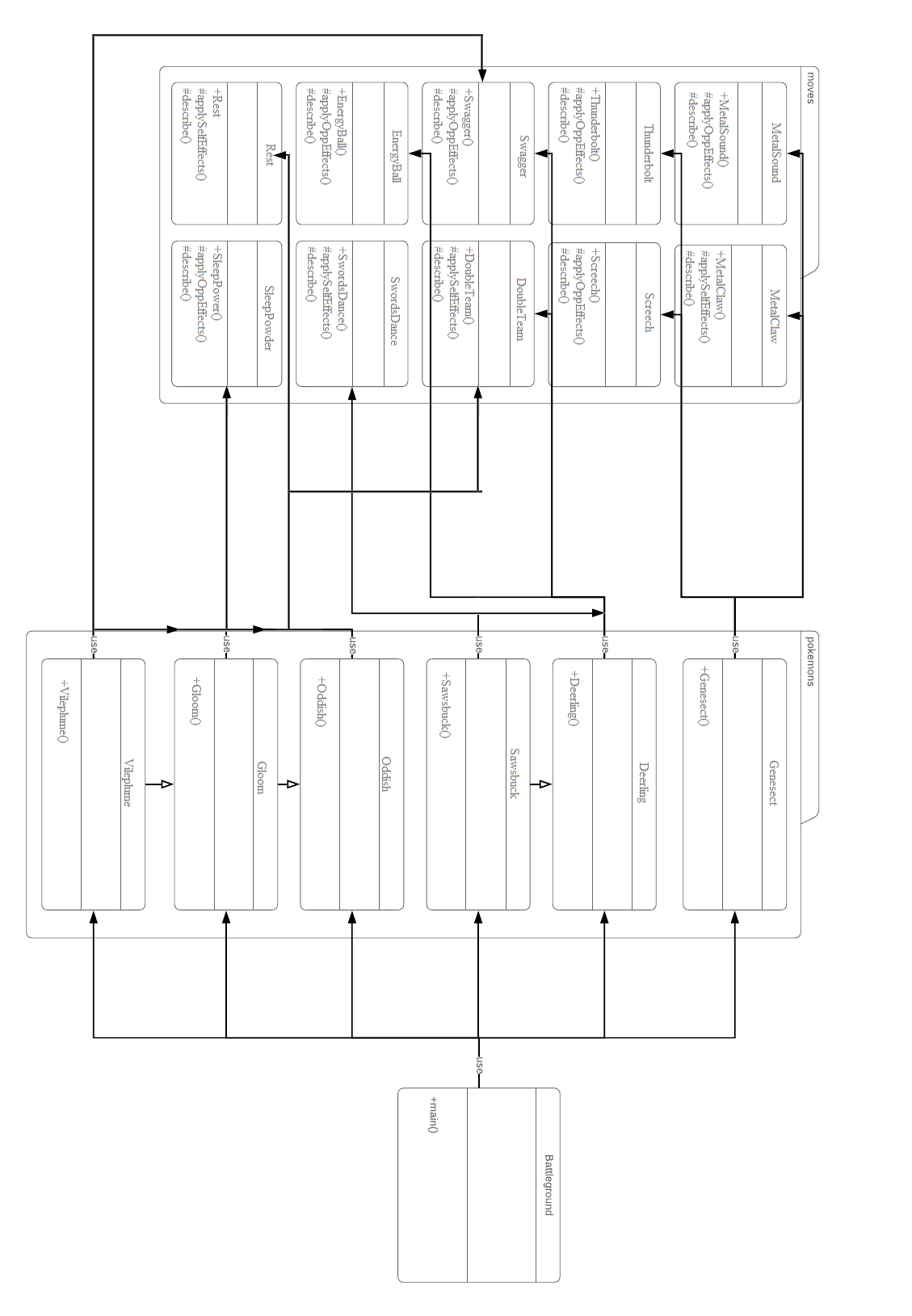
На основе базового класса Pokemon написать свои классы для заданных видов покемонов. Каждый вид покемона должен иметь один или два типа и стандартные базовые характеристики:

* очки здоровья (HP)
* атака (attack)
* защита (defense)
* специальная атака (special attack)
* специальная защита (special defense)
* скорость (speed)

## Покемоны:



## Диаграмма классов:



# Кодпрограммы:

## **Battleground.java**

import pokemons.\*;

import ru.ifmo.se.pokemon.\*;

public class Battleground {

public static void main(String[] args) {

Battle b = new Battle();

Genesect p1 = new Genesect("number1", 1);

Deerling p2 = new Deerling("number 2",1);

Sawsbuck p3 = new Sawsbuck("number 3", 1);

Oddish p4 = new Oddish("number 4", 1);

Gloom p5 = new Gloom("number 5", 1);

Vileplume p6 = new Vileplume("number 6", 1);

b.addAlly(p1);

b.addAlly(p2);

b.addAlly(p3);

b.addFoe(p4);

b.addFoe(p5);

b.addFoe(p6);

b.go();

}

}

# Коды покемон:

## **Genesect.java**

package pokemons;

import moves.\*;

import ru.ifmo.se.pokemon.\*;

public class Genesect extends Pokemon {

public Genesect(String name, int level) {

super(name, level);

setStats(71, 120, 95, 120, 95, 99);

setType(Type.BUG, Type.STEEL);

setMove(new MetalSound(), new MetalClaw(), new Thunderbolt(), new Screech());

}

}

## **Deerling.java**

package pokemons;

import moves.\*;

import ru.ifmo.se.pokemon.\*;

public class Deerling extends Pokemon {

public Deerling(String name, int level) {

super(name, level);

setStats(60, 60, 50, 40, 50, 75);

setType(Type.NORMAL, Type.GRASS);

setMove(new Swagger(), new EnergyBall(), new DoubleTeam());

}

}

## **Sawbuck.java**

package pokemons;

import moves.\*;

public class Sawsbuck extends Deerling {

public Sawsbuck(String name, int level) {

super(name, level);

setStats(80, 100, 70, 60, 70, 95);

addMove(new SwordsDance());

}

}

## **Oddish.java**

package pokemons;

import moves.\*;

import ru.ifmo.se.pokemon.\*;

public class Oddish extends Pokemon{

public Oddish(String name, int level) {

super(name, level);

setStats(45, 50, 55, 75, 65, 30);

setType(Type.GRASS, Type.POISON);

setMove(new DoubleTeam(), new Rest(getHP()));

}

}

## **Gloom.java**

package pokemons;

import moves.\*;

public class Gloom extends Oddish {

public Gloom(String name, int level) {

super(name, level);

setStats(60, 65, 70, 85, 75, 40);

addMove(new SleepPowder());

}

}

## **Vileplume.java**

package pokemons;

import moves.\*;

public class Vileplume extends Gloom{

public Vileplume(String name, int level) {

super(name, level);

setStats(75, 80, 85, 110, 90, 50);

addMove(new Swagger());

}

}

# Коды move:

## **DoubleTeam.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class DoubleTeam extends StatusMove {

public DoubleTeam() {

super(Type.NORMAL, 0, 0);

}

@Override

protected void applySelfEffects(Pokemon p) {

if (p.getStat(Stat.EVASION) < 6)

p.setMod(Stat.EVASION, 1);

}

@Override

protected String describe() {

return "использует Double Team";

}

}

## **EnergyBall.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class EnergyBall extends SpecialMove {

public EnergyBall() {

super(Type.GRASS, 90, 100);

}

@Override

protected void applyOppEffects(Pokemon p) {

if(Math.random() < 0.1) {

if (p.getStat(Stat.SPECIAL\_DEFENSE) > -6) {

p.setMod(Stat.SPECIAL\_DEFENSE, -1);

}

}

}

@Override

protected String describe() {

return "использует Energy Ball";

}

}

## **MetalClaw.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class MetalClaw extends PhysicalMove {

public MetalClaw() {

super(Type.STEEL, 50, 95);

}

@Override

protected void applySelfEffects(Pokemon p) {

if (Math.random() < 0.1) {

if (p.getStat(Stat.ATTACK) < 6) {

p.setMod(Stat.ATTACK, 1);

}

}

}

@Override

protected String describe() {

return "использует Metal Claw";

}

}

## **MetalSound.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class MetalClaw extends PhysicalMove {

public MetalClaw() {

super(Type.STEEL, 50, 95);

}

@Override

protected void applySelfEffects(Pokemon p) {

if (Math.random() < 0.1) {

if (p.getStat(Stat.ATTACK) < 6) {

p.setMod(Stat.ATTACK, 1);

}

}

}

@Override

protected String describe() {

return "использует Metal Claw";

}

}

## **Rest.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class Rest extends StatusMove {

double maxHP=0;

public Rest(double maxHP) {

super(Type.PSYCHIC, 0, 0);

this.maxHP=maxHP;

}

@Override

protected void applySelfEffects(Pokemon p) {

Effect e1 = new Effect().turns(0).stat(Stat.HP, (int) (maxHP-p.getHP()));

Effect e2 = new Effect().turns(2).condition(Status.SLEEP);

p.addEffect(e1);

p.setCondition(e2);

}

@Override

protected String describe() {

return "использует Rest";

}

}

## **Screech.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class Screech extends StatusMove {

public Screech() {

super(Type.NORMAL, 0, 85);

}

@Override

protected void applyOppEffects(Pokemon p) {

if(p.getStat(Stat.DEFENSE) > -5)

p.setMod(Stat.DEFENSE, -2);

}

@Override

protected String describe() {

return "использует Screech";

}

}

## **SleepPowder.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class SleepPowder extends StatusMove{

public SleepPowder() {

super(Type.GRASS, 0, 75);

}

@Override

protected void applyOppEffects(Pokemon p) {

Effect.sleep(p);

}

@Override

protected String describe() {

return "использует Sleep Powder";

}

}

## **Swagger.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class Swagger extends StatusMove {

public Swagger() {

super(Type.NORMAL, 0, 85);

}

@Override

protected void applyOppEffects(Pokemon p) {

if (p.getStat(Stat.ATTACK) < 5)

p.setMod(Stat.ATTACK, 2);

Effect.confuse(p);

}

@Override

protected String describe() {

return "использует Swagger";

}

}

## **SwordsDance.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class SwordsDance extends StatusMove {

public SwordsDance() {

super(Type.NORMAL, 0, 0);

}

@Override

protected void applySelfEffects(Pokemon p) {

if (p.getStat(Stat.ATTACK) <5)

p.setMod(Stat.ATTACK, 2);

}

@Override

protected String describe() {

return "использует Swords Dance";

}

}

## **Thunderbolt.java**

package moves;

import ru.ifmo.se.pokemon.\*;

public class Thunderbolt extends SpecialMove{

public Thunderbolt() {

super(Type.ELECTRIC, 90, 100);

}

@Override

protected void applyOppEffects(Pokemon p) {

if(Math.random() < 0.1) {

Effect.paralyze(p);

}

}

@Override

protected String describe() {

return "использует Thunderbolt";

}

# Результат работы:

Genesect number1 from the team Yellow enters the battle!

Oddish number 4 from the team Black enters the battle!

Genesect number1 использует Thunderbolt.

Oddish number 4 loses 2 hit points.

Oddish number 4 misses

Genesect number1 использует Screech.

Oddish number 4 decreases defense.

Oddish number 4 misses

Genesect number1 использует Metal Claw.

Oddish number 4 loses 4 hit points.

Oddish number 4 misses

Genesect number1 использует Thunderbolt.

Oddish number 4 loses 3 hit points.

Oddish number 4 misses

Genesect number1 использует Thunderbolt.

Oddish number 4 loses 2 hit points.

Oddish number 4 misses

Genesect number1 использует Screech.

Oddish number 4 decreases defense.

Oddish number 4 misses

Genesect number1 использует Thunderbolt.

Oddish number 4 loses 2 hit points.

Oddish number 4 is paralyzed

Oddish number 4 faints.

Gloom number 5 from the team Black enters the battle!

Genesect number1 использует Metal Claw.

Gloom number 5 loses 5 hit points.

Gloom number 5 использует Sleep Powder.

Genesect number1 is sleeping

Gloom number 5 использует Sleep Powder.

Genesect number1 использует Metal Sound.

Gloom number 5 decreases special defense.

Gloom number 5 использует Sleep Powder.

Genesect number1 is sleeping

Gloom number 5 misses

Genesect number1 использует Thunderbolt.

Gloom number 5 loses 2 hit points.

Gloom number 5 использует Sleep Powder.

Genesect number1 is sleeping

Gloom number 5 misses

Gloom number 5 misses

Genesect number1 использует Metal Claw.

Gloom number 5 loses 5 hit points.

Gloom number 5 использует Sleep Powder.

Genesect number1 is sleeping

Genesect number1 использует Metal Claw.

Gloom number 5 loses 9 hit points.

Gloom number 5 faints.

Vileplume number 6 from the team Black enters the battle!

Genesect number1 использует Screech.

Vileplume number 6 decreases defense.

Vileplume number 6 использует Sleep Powder.

Genesect number1 is sleeping

Vileplume number 6 misses

Vileplume number 6 misses

Genesect number1 использует Metal Sound.

Vileplume number 6 decreases special defense.

Vileplume number 6 misses

Genesect number1 использует Metal Claw.

Vileplume number 6 loses 5 hit points.

Vileplume number 6 использует Sleep Powder.

Genesect number1 is sleeping

Vileplume number 6 использует Swagger.

Vileplume number 6 misses

Genesect number1 использует Metal Sound.

Vileplume number 6 decreases special defense.

Vileplume number 6 misses

Genesect number1 использует Screech.

Vileplume number 6 decreases defense.

Vileplume number 6 misses

Genesect number1 использует Thunderbolt.

Vileplume number 6 loses 4 hit points.

Vileplume number 6 использует Swagger.

Genesect number1 использует Metal Sound.

Vileplume number 6 decreases special defense.

Vileplume number 6 использует Swagger.

Genesect number1 hits himself in confusion.

Genesect number1 loses 5 hit points.

Vileplume number 6 misses

Genesect number1 использует Metal Claw.

Vileplume number 6 loses 11 hit points.

Vileplume number 6 faints.

Team Black loses its last Pokemon.

The team Yellow wins the battle!

# Вывод:

Во время выполнения данной лабораторной роботы я изучил основные принципы ООП, научился работать с методами, классами, модификаторами доступами и сторонними библиотеками.