



**ITMO UNIVERSITY**

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики»

**Факультет Систем управления и робототехники**

**Лабораторная работа №2**

по дисциплине

«ПРОГРАММИРОВАНИЕ»

Вариант №30234

**Выполнил:**

Студент группы R3142

Дильмиев Тимур Рашитович

**Преподаватель:**

Перцев Тимофей Сергеевич

Санкт-Петербург

## 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)

Классы покемонов должны наследоваться в соответствии с цепочкой эволюции покемонов. На основе базовых классов **PhysicalMove**, **SpecialMove** и **StatusMove** реализовать свои классы для заданных видов атак.

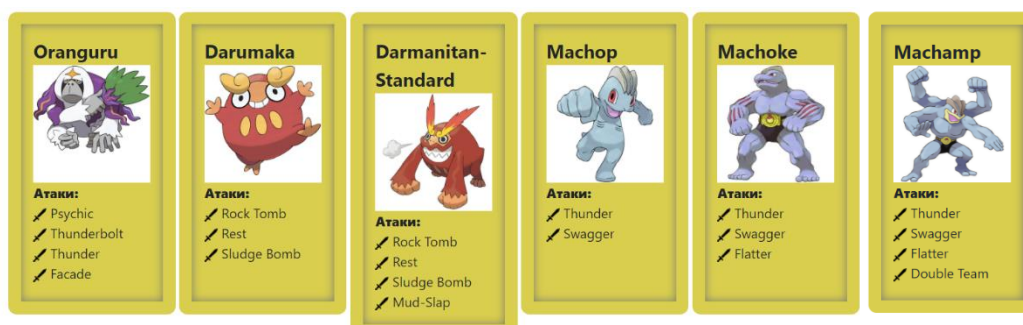
Атака должна иметь стандартные тип, силу (power) и точность (accuracy). Должны быть реализованы стандартные эффекты атаки. Назначить каждому виду покемонов атаки в соответствии с вариантом. Уровень покемона выбирается минимально необходимым для всех реализованных атак.

Используя класс симуляции боя **Battle**, создать 2 команды покемонов (каждый покемон должен иметь имя) и запустить бой.

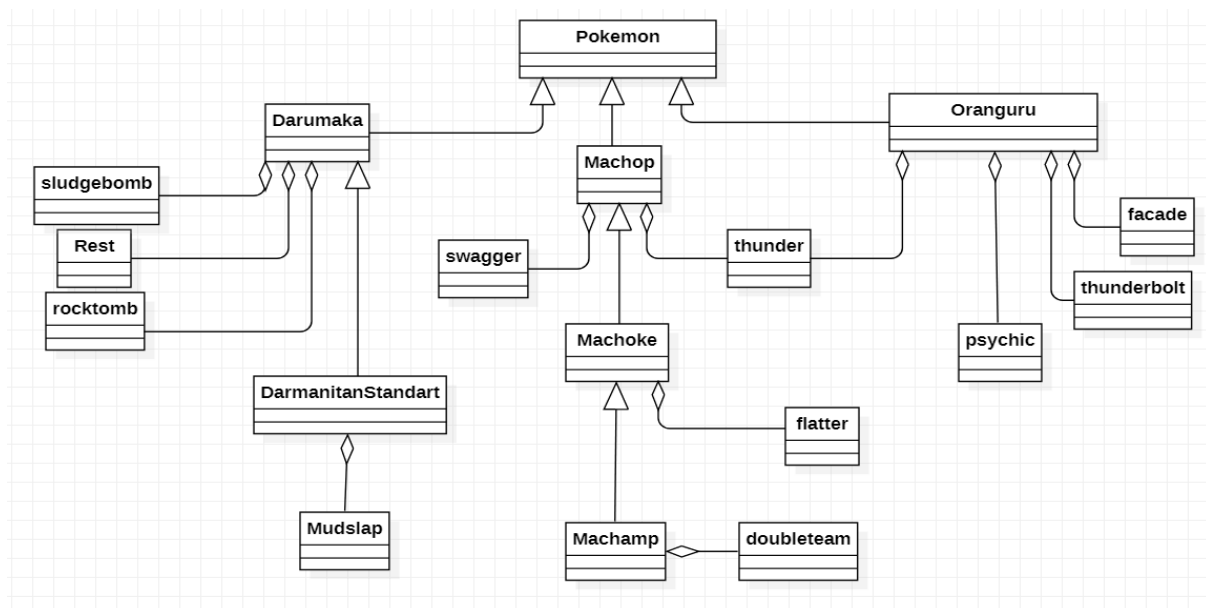
Базовые классы и симулятор сражения находятся в [jar-архиве](#) (обновлен 9.10.2018, исправлен баг с добавлением атак и кодировкой). Документация в формате javadoc - [здесь](#).

Информацию о покемонах, цепочках эволюции и атаках можно найти на сайтах <http://poke-universe.ru>, <http://pokemondb.net>, <http://veekun.com/dex/pokemon>

## ПОКЕМОНЫ:



## ДИАГРАММА КЛАССОВ:



## ИСХОДНЫЙ КОД:

Battleground.java

```
1
2 import ru.ifmo.se.pokemon.*;
3 public final class Battleground {
4     public static void main(String[] args){
5         Battle c = new Battle();
6         Pokemon p1 = new Oranguru( name: "lox" , level: 1);
7         Pokemon p2 = new Darumaka( name: "cop" , level: 1);
8         Pokemon p3 = new DarmanitanStandart( name: "big" , level: 2);
9         Pokemon p4 = new Machop( name: "dok" , level: 1);
10        Pokemon p5 = new Machoke( name: "sir" , level: 2);
11        Pokemon p6 = new Machamp( name: "oct" , level: 3);
12        c.addAlly(p1);
13        c.addAlly(p2);
14        c.addAlly(p3);
15        c.addFoe(p4);
16        c.addFoe(p5);
17        c.addFoe(p6);
18        c.go();
19    }
20 }
```

## DarmanitanStandart.java

```
1
2 import ru.ifmo.se.pokemon.*;
3 public class DarmanitanStandart extends Darumaka {
4     public DarmanitanStandart(String name, int level){
5         super(name, level);
6         setStats(v: 105, v1: 140, v2: 55, v3: 30, v4: 55, v5: 95);
7         setType(Type.GROUND, Type.PSYCHIC, Type.ROCK, Type.POISON);
8         setMove(new RockTomb(),new Rest(),new SludgeBomb(),new Mudslap());
9     }
10 }
11 |
```

## Darumaka.java

```
1
2 import ru.ifmo.se.pokemon.*;
3 public class Darumaka extends Pokemon{
4     public Darumaka(String name, int level){
5         super(name, level);
6         setStats(v: 70, v1: 90, v2: 45, v3: 15, v4: 45, v5: 50);
7         setType(Type.PSYCHIC, Type.ROCK, Type.POISON);
8         setMove(new RockTomb(),new Rest(),new SludgeBomb());
9     }
10 }
11 |
```

## Machop.java

```
1
2 import ru.ifmo.se.pokemon.*;
3 public class Machop extends Pokemon{
4     public Machop(String name, int level){
5         super(name, level);
6         setStats(v: 70, v1: 80, v2: 50, v3: 35, v4: 35, v5: 35);
7         setType(Type.ELECTRIC, Type.NORMAL);
8         setMove(new Thunder(),new Swagger());
9     }
10 }
11 |
```

## Machoke.java

```
1
2 import ru.ifmo.se.pokemon.*;
3 public class Machoke extends Machop{
4     public Machoke(String name, int level){
5         super(name, level);
6         setStats(v: 80, v1: 100, v2: 70, v3: 50, v4: 60, v5: 45);
7         setType(Type.ELECTRIC, Type.NORMAL, Type.DARK);
8         setMove(new Thunder(),new Swagger(), new Flatter());
9     }
10 }
11 |
```

## Machamp.java

```
1
2  import ru.ifmo.se.pokemon.*;
3  public class Machamp extends Machoke{
4      public Machamp(String name, int level){
5          super(name, level);
6          setStats(v: 90, v1: 130, v2: 80, v3: 65, v4: 85, v5: 55);
7          setType(Type.ELECTRIC, Type.NORMAL, Type.DARK);
8          setMove(new Thunder(),new Swagger(), new Flatter(),new DoubleTeam());
9      }
10 }
```

## Oranguru.java

```
1
2
3  import ru.ifmo.se.pokemon.*;
4  public class Oranguru extends Pokemon {
5      public Oranguru(String name, int level) {
6          super(name, level);
7          setStats(v: 90, v1: 60, v2: 80, v3: 90, v4: 110, v5: 60);
8          setType(Type.ELECTRIC, Type.PSYCHIC, Type.NORMAL);
9          setMove(new Psychic(), new Thunderbolt(), new Thunder(), new Facade());
10     }
11 }
```

## DoubleTeam.java

```
1
2  import ru.ifmo.se.pokemon.Pokemon;
3  import ru.ifmo.se.pokemon.Stat;
4  import ru.ifmo.se.pokemon.StatusMove;
5  import ru.ifmo.se.pokemon.Type;
6
7  class DoubleTeam extends StatusMove {
8      protected DoubleTeam(){
9          super(Type.NORMAL, v: 0, v1: 0);
10     }
11     @Override
12     protected void applySelfEffects(Pokemon p){
13         p.setMod(Stat.EVASION, E 1);
14     }
15     @Override
16     protected String describe() {
17         return "увеличивается уклончивость на 1";
18     }
19 }
```

## Swagger.java

```
1
2  import ru.ifmo.se.pokemon.*;
3
4  class Swagger extends StatusMove {
5      protected Swagger() {
6          super(Type.NORMAL, v: 0, v1: 85);
7      }
8      @Override
9      protected void applyOppEffects(Pokemon p) {
10         p.setMod(Stat.ATTACK, E 2);
11         Effect.confuse(p);
12     }
13 }
```

## Mudslap.java

```
1
2 import ru.ifmo.se.pokemon.Pokemon;
3 import ru.ifmo.se.pokemon.SpecialMove;
4 import ru.ifmo.se.pokemon.Stat;
5 import ru.ifmo.se.pokemon.Type;
6
7 class Mudslap extends SpecialMove {
8     protected Mudslap() {
9         super(Type.GROUND, 20, 100);
10    }
11
12    @Override
13    protected void applyOppEffects(Pokemon p) {
14        p.setMod(Stat.SPEED, -1);
15    }
16
17    @Override
18    protected String describe() {
19        return "наносит урон и понижает точность противника на 1 пункт";
20    }
21 }
```

## Facade.java

```
1
2 import ru.ifmo.se.pokemon.*;
3
4 class Facade extends SpecialMove {
5     protected Facade() {
6         super(Type.NORMAL, 20, 70);
7     }
8
9     @Override
10    protected void applyOppDamage(Pokemon def, double damage) {
11        Status Pokcon = def.getCondition();
12        if (Pokcon.equals(Status.BURN) || Pokcon.equals(Status.POISON) ||
13            Pokcon.equals(Status.PARALYZE)) {
14            def.setMod(Stat.HP, (int) Math.round(damage) * 2);
15        }
16    }
17
18    @Override
19    protected String describe() {
20        return "сила удваивается, если использующий горит, парализован или отравлен";
21    }
22 }
```

## Flutter.java

```
1 import ru.ifmo.se.pokemon.*;
2
3 class Flutter extends StatusMove {
4     protected Flutter() {
5         super(Type.DARK, 0, 100);
6     }
7
8     @Override
9     protected void applyOppEffects(Pokemon p) {
10        Effect.confuse(p);
11        p.setMod(Stat.SPECIAL_ATTACK, +2);
12    }
13
14    @Override
15    protected String describe() {
16        return "сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня";
17    }
18 }
```

## Psychic.java

```
1
2 import ru.ifmo.se.pokemon.*;
3 import java.lang.Math;
4 class Psychic extends SpecialMove{
5     protected Psychic(){
6         super(Type.PSYCHIC, 10, 90);
7     }
8     @Override
9     protected void applyOppEffects(Pokemon p){
10         if (Math.random()<=0.1)
11             p.setMod(Stat.SPECIAL_DEFENSE, -1);
12     }
13     @Override
14     protected String describe(){
15         return "uses Psychic";
16     }
17 }
```

## Rest.java

```
1
2 import ru.ifmo.se.pokemon.*;
3
4 class Rest extends StatusMove {
5     protected Rest() {
6         super(Type.PSYCHIC, 0, 0);
7     }
8
9     @Override
10    protected void applySelfEffects(Pokemon p) {
11        Effect.sleep(p);
12        p.setMod(Stat.HP, +1000);
13    }
14
15    @Override
16    protected String describe() {
17        return "пропускает 2 хода, но полностью восстанавливает здоровье";
18    }
19 }
```

## Rocktomb.java

```
1
2 import ru.ifmo.se.pokemon.Pokemon;
3 import ru.ifmo.se.pokemon.SpecialMove;
4 import ru.ifmo.se.pokemon.Stat;
5 import ru.ifmo.se.pokemon.Type;
6
7 class RockTomb extends SpecialMove {
8     protected RockTomb() {
9         super(Type.ROCK, 60, 95);
10    }
11
12    @Override
13    protected void applyOppEffects(Pokemon p) {
14        p.setMod(Stat.SPEED, -1);
15    }
16
17    @Override
18    protected String describe() {
19        return "наносит урон и понижает скорость противника на 1 пункт";
20    }
21 }
```

## SludgeBomb.java

```
1
2 import ru.ifmo.se.pokemon.Effect;
3 import ru.ifmo.se.pokemon.Pokemon;
4 import ru.ifmo.se.pokemon.SpecialMove;
5 import ru.ifmo.se.pokemon.Type;
6 class SludgeBomb extends SpecialMove {
7     protected SludgeBomb() {
8         super(Type.POISON, 90, 100);
9     }
10
11     @Override
12     protected void applyOppEffects(Pokemon p) {
13         if (Math.random() <= 0.3) Effect.poison(p);
14     }
15
16     @Override
17     protected String describe() {
18         return "наносит урон и есть 30% шанс отравить противника";
19     }
20 }
21 }
```

## Thunder.java

```
1
2 import ru.ifmo.se.pokemon.Effect;
3 import ru.ifmo.se.pokemon.Pokemon;
4 import ru.ifmo.se.pokemon.SpecialMove;
5 import ru.ifmo.se.pokemon.Type;
6
7 class Thunder extends SpecialMove {
8     protected Thunder() {
9         super(Type.ELECTRIC, 10, 110);
10    }
11
12    @Override
13    protected void applyOppEffects(Pokemon p) {
14        if (Math.random() <= 0.3) Effect.paralyze(p);
15    }
16
17    @Override
18    protected String describe() {
19        return "наносит урон, есть 30% шанс парализовать противника";
20    }
21 }
```

## Thunderbolt.java

```
1
2 import ru.ifmo.se.pokemon.Effect;
3 import ru.ifmo.se.pokemon.Pokemon;
4 import ru.ifmo.se.pokemon.SpecialMove;
5 import ru.ifmo.se.pokemon.Type;
6
7 class Thunderbolt extends SpecialMove {
8     protected Thunderbolt() {
9         super(Type.ELECTRIC, 15, 90);
10    }
11
12    @Override
13    protected void applyOppEffects(Pokemon p) {
14        if (Math.random() <= 0.1) Effect.paralyze(p);
15    }
16
17    @Override
18    protected String describe() {
19        return "наносит урон, есть 10% шанс парализовать противника";
20    }
21 }
```



## Результат работы:

Oranguru lox из команды фиолетовых вступает в бой!

Machop dok из команды красных вступает в бой!

Oranguru lox наносит урон, есть 10% шанс парализовать противника.

Machop dok теряет 2 здоровья.

Machop dok наносит урон, есть 30% шанс парализовать противника.

Oranguru lox теряет 3 здоровья.

Oranguru lox uses Psychic.

Machop dok теряет 5 здоровья.

Machop dok уменьшает специальную защиту.

Machop dok атакует.

Oranguru lox увеличивает атаку.

Oranguru lox сила удваивается, если использующий горит, парализован или отравлен.

Machop dok атакует.

Oranguru lox увеличивает атаку.

Oranguru lox наносит урон, есть 30% шанс парализовать противника.

Machop dok теряет 3 здоровья.

Machop dok атакует.

Oranguru lox увеличивает атаку.

Oranguru lox наносит урон, есть 30% шанс парализовать противника.

Machop dok теряет 3 здоровья.

Machop dok теряет сознание.

Machoke sir из команды красных вступает в бой!

Machoke sir атакует.

Oranguru lox увеличивает атаку.

Oranguru lox uses Psychic.

Machoke sir теряет 5 здоровья.

Machoke sir сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня.

Oranguru lox увеличивает специальную атаку.

Oranguru lox наносит урон, есть 10% шанс парализовать противника.

Machoke sir теряет 2 здоровья.

Machoke sir атакует.

Oranguru lox увеличивает атаку.

Oranguru lox растерянно попадает по себе.

Oranguru lox теряет 4 здоровья.

Machoke sir сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня.

Oranguru lox увеличивает специальную атаку.

Oranguru lox наносит урон, есть 10% шанс парализовать противника.

Machoke sir теряет 2 здоровья.

Machoke sir сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня.

Oranguru lox увеличивает специальную атаку.

Oranguru lox растерянно попадает по себе.

Oranguru lox теряет 5 здоровья.

Machoke sir наносит урон, есть 30% шанс парализовать противника.

Oranguru lox теряет 3 здоровья.

Oranguru lox теряет сознание.

Darumaka сор из команды фиолетовых вступает в бой!

Machoke sir сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня.

Darumaka сор увеличивает специальную атаку.

Darumaka сор промахивается

Machoke sir сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня.

Darumaka сор увеличивает специальную атаку.

Darumaka сор наносит урон и есть 30% шанс отравить противника.

Machoke sir теряет 6 здоровья.

Machoke sir отравлен

Machoke sir теряет 1 здоровья.

Machamp ост из команды красных вступает в бой!

Machamp ост промахивается

Darumaka сор наносит урон и есть 30% шанс отравить противника.

Критический удар!

Machamp ост теряет 6 здоровья.

Machamp ост сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня.

Darumaka сор увеличивает специальную атаку.

Darumaka сор наносит урон и есть 30% шанс отравить противника.

Machamp ост теряет 5 здоровья.

Machamp ост сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня.

Darumaka сор увеличивает специальную атаку.

Darumaka сор промахивается

Machamp ост сбивает с толку противника, но специальная атака противника увеличивается на 2 уровня.

Darumaka сор увеличивает специальную атаку.

Darumaka сор наносит урон и понижает скорость противника на 1 пункт.

Machamp ост теряет 7 здоровья.

Machamp ост уменьшает скорость.

Machamp ост атакует.

Darumaka сор увеличивает атаку.

Darumaka сор наносит урон и есть 30% шанс отравить противника.

Machamp ост теряет 4 здоровья.

Machamp ост отравлен

Machamp ост теряет сознание.

В команде красных не осталось покемонов.

Команда фиолетовых побеждает в этом бою!

Process finished with exit code 0

### Вывод:

В результате выполнения лабораторной работы были получены навыки в объектно-ориентированном программировании при использовании языка Java.