Санкт-Петербургский Национальный Исследовательский

Университет Информационных технологий, Механики и оптики

Факультет Систем Управления и Робототехники



Вариант № 311799

Лабораторная работа №2

По дисциплине:

ПРОГРАММИРОВАНИЕ

Выполнила:

Студент группы Р3117 Демьянова Анастасия

Преподаватель: Письмак Алексей Евгеньевич

Санкт-Петербург

2022 г.

# Оглавление

[Оглавление 2](#_Toc99702216)

[Текст задания 3](#_Toc99702217)

[Диаграмма классов 4](#_Toc99702218)

[Исходный код 5](#_Toc99702219)

[Main 5](#_Toc99702220)

[Tyrogue 5](#_Toc99702221)

[Shuckle 5](#_Toc99702222)

[Metang 6](#_Toc99702223)

[Metagross 6](#_Toc99702224)

[Hitmonlee 7](#_Toc99702225)

[Beldum 7](#_Toc99702226)

[ThunderShock 7](#_Toc99702227)

[RockSlide 8](#_Toc99702228)

[Rest 8](#_Toc99702229)

[Psychic 8](#_Toc99702230)

[PoisonJab 8](#_Toc99702231)

[IceBeam 9](#_Toc99702232)

[Facade 9](#_Toc99702233)

[DynamicPunch 9](#_Toc99702234)

[DualChop 9](#_Toc99702235)

[Confide 10](#_Toc99702236)

[Результат исполнения программы на сервере Helios 11](#_Toc99702237)

[Вывод 13](#_Toc99702238)

# Текст задания

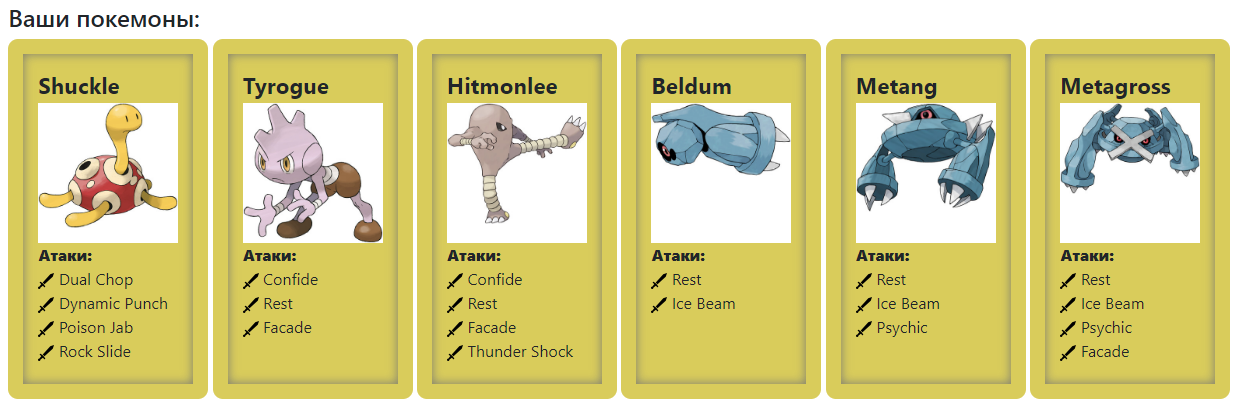
На основе базового класса Pokemon написать свои классы для заданных видов покемонов. Каждый вид покемона должен иметь один или два типа и стандартные базовые характеристики:

* 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.



# Диаграмма классов

# Исходный код

## Main

import Pokemons.\*;  
import ru.ifmo.se.pokemon.\*;  
  
public class Main {  
 public static void main(String[] args) {  
 Battle battle = new Battle();  
 battle.addAlly(new Shuckle("SHuK", 2));  
 battle.addAlly(new Beldum("BelD", 1));  
 battle.addAlly(new Metagross("MetaGross", 3));  
 battle.addFoe(new Hitmonlee("Hitman", 2));  
 battle.addFoe(new Metang("Meta", 3));  
 battle.addFoe(new Tyrogue("TiRoga", 1));  
 battle.go();  
 }  
}

## Tyrogue

package Pokemons;  
import ru.ifmo.se.pokemon.\*;  
import Moves.\*;  
  
public class Tyrogue extends Pokemon {  
 public final Type pt1 = Type.*DRAGON* , pt2 = Type.*NORMAL*;  
 final double HP = 78;  
 final double attack = 60;  
 final double defense = 85;  
 final double special\_attack = 135;  
 final double special\_defense = 91;  
 final double speed = 36;  
 public Tyrogue(final String name, final int level) {  
 super(name, level);  
 this.setStats(HP, attack, defense, special\_attack, special\_defense, speed);  
 this.setMove(new Facade(pt1, 10, 50));  
 this.addMove(new Confide(pt1, 20, 54));  
 this.addMove(new Rest(pt1, 30, 20));  
 }  
}

## Shuckle

package Pokemons;  
import ru.ifmo.se.pokemon.\*;  
import Moves.\*;  
// Statistics are wrong, need to rewrite  
public class Shuckle extends Pokemon {  
 public final Type pt1 = Type.*DRAGON* , pt2 = Type.*NORMAL*;  
 final double HP = 78;  
 final double attack = 60;  
 final double defense = 85;  
 final double special\_attack = 135;  
 final double special\_defense = 91;  
 final double speed = 36;  
 public Shuckle(final String name, final int level) {  
 super(name, level);  
 this.setStats(HP, attack, defense, special\_attack, special\_defense, speed);  
 this.setMove(new DualChop(pt1, 10, 50));  
 this.addMove(new DynamicPunch(pt1, 20, 54));  
 this.addMove(new PoisonJab(pt1, 30, 20));  
 this.addMove(new RockSlide(pt2, 50, 40));  
 }  
}

## Metang

package Pokemons;  
import Moves.IceBeam;  
import Moves.Psychic;  
import Moves.Rest;  
import ru.ifmo.se.pokemon.Type;  
  
public class Metang extends Beldum {  
 public final Type pt1 = Type.*DRAGON* , pt2 = Type.*NORMAL*;  
 final double HP = 78;  
 final double attack = 60;  
 final double defense = 85;  
 final double special\_attack = 135;  
 final double special\_defense = 91;  
 final double speed = 36;  
 public Metang(final String name, final int level) {  
 super(name, level);  
 this.setStats(HP, attack, defense, special\_attack, special\_defense, speed);  
 this.setMove(new Rest(pt1, 10, 50));  
 this.addMove(new IceBeam(pt1, 20, 54));  
 this.addMove(new Psychic(pt1, 20, 54));  
 }  
}

## Metagross

package Pokemons;  
import Moves.Facade;  
import Moves.IceBeam;  
import Moves.Psychic;  
import Moves.Rest;  
import ru.ifmo.se.pokemon.Type;  
  
public class Metagross extends Metang {  
 public final Type pt1 = Type.*DRAGON* , pt2 = Type.*NORMAL*;  
 final double HP = 78;  
 final double attack = 60;  
 final double defense = 85;  
 final double special\_attack = 120;  
 final double special\_defense = 20;  
 final double speed = 36;  
 public Metagross(final String name, final int level) {  
 super(name, level);  
 this.setStats(HP, attack, defense, special\_attack, special\_defense, speed);  
 this.setMove(new Rest(pt1, 10, 50));  
 this.addMove(new IceBeam(pt1, 20, 54));  
 this.addMove(new Psychic(pt1, 20, 54));  
 this.addMove(new Facade(pt1, 20, 54));  
 }  
}

## Hitmonlee

package Pokemons;  
  
import Moves.Confide;  
import Moves.Facade;  
import Moves.Rest;  
import Moves.ThunderShock;  
import ru.ifmo.se.pokemon.Type;  
  
public class Hitmonlee extends Tyrogue {  
 public final Type pt1 = Type.*DRAGON* , pt2 = Type.*NORMAL*;  
 final double HP = 78;  
 final double attack = 60;  
 final double defense = 85;  
 final double special\_attack = 135;  
 final double special\_defense = 91;  
 final double speed = 36;  
 public Hitmonlee(final String name, final int level) {  
 super(name, level);  
 this.setStats(HP, attack, defense, special\_attack, special\_defense, speed);  
 this.setMove(new Facade(pt1, 10, 50));  
 this.addMove(new Confide(pt1, 20, 54));  
 this.addMove(new Rest(pt1, 30, 20));  
 this.addMove(new ThunderShock(pt2, 20, 45));  
 }  
}

## Beldum

package Pokemons;  
import Moves.\*;  
import ru.ifmo.se.pokemon.\*;  
  
public class Beldum extends Pokemon {  
 public final Type pt1 = Type.*DRAGON* , pt2 = Type.*NORMAL*;  
 final double HP = 78;  
 final double attack = 60;  
 final double defense = 85;  
 final double special\_attack = 135;  
 final double special\_defense = 91;  
 final double speed = 36;  
 public Beldum(final String name, final int level) {  
 super(name, level);  
 this.setStats(HP, attack, defense, special\_attack, special\_defense, speed);  
 this.setMove(new Rest(pt1, 10, 50));  
 this.addMove(new IceBeam(pt1, 20, 54));  
 }  
}

## ThunderShock

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class ThunderShock extends SpecialMove {  
 public ThunderShock(final Type type,  
 double pow,  
 double acc) {  
 super(type, pow, acc);  
 }  
  
 protected void applyOppEffect(Pokemon p) {  
 Effect.*confuse*(p);  
 }  
}

## RockSlide

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class RockSlide extends PhysicalMove {  
 public RockSlide(final Type type,  
 double pow,  
 double acc) {  
 super(type, pow, acc);  
 }  
  
 protected void applyOppEffect(Pokemon p) {  
 Effect e = new Effect().chance(0.5).turns(1).stat(Stat.*SPEED*, -1);  
 }  
}

## Rest

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class Rest extends SpecialMove {  
 public Rest(final Type type, double pow, double acc) {  
 super(type, pow, acc);  
 }  
  
 protected void applySelfEffects(Pokemon p){  
 p.setMod(Stat.*HP*, -1);  
 Effect.*sleep*(p);  
 }  
 }

## Psychic

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class Psychic extends SpecialMove {  
 public Psychic(final Type type,  
 double pow,  
 double acc) {  
 super(type, pow, acc);  
 }  
}

## PoisonJab

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class PoisonJab extends SpecialMove {  
 public PoisonJab(final Type type,  
 double pow,  
 double acc) {  
 super(type, pow, acc);  
 }  
}

## IceBeam

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class IceBeam extends SpecialMove {  
 public IceBeam(final Type type,  
 double pow,  
 double acc) {  
 super(type, pow, acc);  
 }  
  
 protected void applyOppEffect(Pokemon p) {  
 Effect.*freeze*(p);  
 }  
}

## Facade

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class Facade extends SpecialMove{  
 public Facade(final Type type,  
 double pow,  
 double acc)  
 {  
 super(type, pow, acc);  
 }  
 protected void applyOppEffects(Pokemon p){  
 Effect.*poison*(p);  
 }  
}

## DynamicPunch

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class DynamicPunch extends SpecialMove {  
 public DynamicPunch(final Type type, double pow, double acc) {  
 super(type, pow, acc);  
 }  
  
 protected void applyOppEffect(Pokemon p) {  
 Effect.*freeze*(p);  
 }  
}

## DualChop

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
  
public class DualChop extends PhysicalMove {  
 public DualChop(final Type type,  
 double pow,  
 double acc)  
 {  
 super(type, pow, acc);  
 }  
  
 protected void applyOppEffects(Pokemon p){  
 Effect.*poison*(p);  
 }  
}

## Confide

package Moves;  
import ru.ifmo.se.pokemon.\*;  
  
public class Confide extends PhysicalMove {  
 public Confide(final Type type,  
 double pow,  
 double acc) {  
 super(type, pow, acc);  
 }  
}

# Результат исполнения программы на сервере Helios

Shuckle SHuK из команды фиолетовых вступает в бой!

Hitmonlee Hitman из команды белых вступает в бой!

Hitmonlee Hitman атакует.

Shuckle SHuK теряет 5 здоровья.

Shuckle SHuK атакует.

Hitmonlee Hitman теряет 4 здоровья.

Shuckle SHuK атакует.

Hitmonlee Hitman теряет 4 здоровья.

Hitmonlee Hitman атакует.

Shuckle SHuK теряет 4 здоровья.

Shuckle SHuK отравлен

Shuckle SHuK теряет 1 здоровья.

Shuckle SHuK атакует.

Hitmonlee Hitman теряет 3 здоровья.

Hitmonlee Hitman атакует.

Shuckle SHuK теряет 3 здоровья.

Shuckle SHuK теряет 1 здоровья.

Shuckle SHuK атакует.

Hitmonlee Hitman теряет 3 здоровья.

Hitmonlee Hitman атакует.

Shuckle SHuK теряет 4 здоровья.

Shuckle SHuK теряет сознание.

Beldum BelD из команды фиолетовых вступает в бой!

Hitmonlee Hitman атакует.

Beldum BelD теряет 3 здоровья.

Beldum BelD атакует.

Hitmonlee Hitman теряет 3 здоровья.

Beldum BelD восстанавливает 1 здоровья.

Beldum BelD засыпает

Hitmonlee Hitman теряет сознание.

Metang Meta из команды белых вступает в бой!

Metang Meta атакует.

Beldum BelD теряет 3 здоровья.

Metang Meta атакует.

Beldum BelD теряет 3 здоровья.

Metang Meta атакует.

Beldum BelD теряет 3 здоровья.

Metang Meta атакует.

Beldum BelD теряет 5 здоровья.

Beldum BelD теряет сознание.

Metagross MetaGross из команды фиолетовых вступает в бой!

Metagross MetaGross атакует.

Metagross MetaGross теряет 5 здоровья.

Metagross MetaGross атакует.

Metagross MetaGross теряет 5 здоровья.

Metagross MetaGross атакует.

Metang Meta теряет 3 здоровья.

Metang Meta атакует.

Metagross MetaGross теряет 3 здоровья.

Metang Meta восстанавливает 1 здоровья.

Metang Meta засыпает

Metagross MetaGross атакует.

Metang Meta теряет 3 здоровья.

Metang Meta отравлен

Metang Meta атакует.

Metagross MetaGross теряет 4 здоровья.

Metang Meta восстанавливает 1 здоровья.

Metang Meta засыпает

Metagross MetaGross атакует.

Metang Meta теряет 4 здоровья.

Metagross MetaGross восстанавливает 1 здоровья.

Metagross MetaGross засыпает

Metagross MetaGross атакует.

Metang Meta теряет 3 здоровья.

Metagross MetaGross атакует.

Metang Meta теряет 3 здоровья.

Metang Meta отравлен

Metang Meta атакует.

Metagross MetaGross теряет 6 здоровья.

Metagross MetaGross теряет сознание.

В команде фиолетовых не осталось покемонов.

Команда белых побеждает в этом бою!

# Вывод

В результате выполнения работы были получены умения реализации своих классов на основе базовых из модуля Pokemon.jar.

В процессе выполнения поставленной задачи была произведена компиляция программ с библиотекой, а затем создание, работающего Jar-архива на сервере.

Команды для решения проблемы (Linux):

* javac -sourcepath src -classpath lib/Pokemon.jar -d out src/[Main.java](https://vk.com/away.php?to=http%3A%2F%2FMain.java&cc_key=)
* java -cp out:lib/Pokemon.jar Main
* echo main-class: Main>manifest.mf
* echo class-path: lib/Pokemon.jar»manifest.mf
* jar -cmfv [manifest.mf](https://vk.com/away.php?to=http%3A%2F%2Fmanifest.mf&cc_key=" \t "_blank) Main.jar -C out .

Команды для решения проблемы (Win):

* javac -sourcepath src -classpath lib\Pokemon.jar -d out src\[Main.java](https://vk.com/away.php?to=http%3A%2F%2FMain.java&cc_key=)
* java -cp out;lib\Pokemon.jar Main

Заполняем файл manifest.mf:

* main-class: Main
* class-path: lib\Pokemon.jar
* jar -cmfv [manifest.mf](https://vk.com/away.php?to=http%3A%2F%2Fmanifest.mf&cc_key=" \t "_blank) Main.jar -C out .