Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики.

Лабораторная работа №2

По дисциплине «Программирование»

Вариант № 32530

Выполнил:

Русинов Дмитрий Станиславович P3132

Преподаватель:

Горбунов Михаил Витальевич

Санкт-Петербург

2021

Оглавление

[Текст задания 3](#_Toc85288407)

[Покемоны и их атаки варианта: 4](#_Toc85288408)

[Исходный код: 6](#_Toc85288409)

[Main 6](#_Toc85288410)

[Bouffalant 6](#_Toc85288411)

[Cosmoem 7](#_Toc85288412)

[Munchlax 7](#_Toc85288413)

[Nigilego 7](#_Toc85288414)

[Snorlax 8](#_Toc85288415)

[Solgaleo 8](#_Toc85288416)

[Aerial\_Ace 8](#_Toc85288417)

[Blizzard 9](#_Toc85288418)

[Body\_Slam 9](#_Toc85288419)

[Charge\_Beam 9](#_Toc85288420)

[Double\_Team 10](#_Toc85288421)

[Facade 10](#_Toc85288422)

[Leer 11](#_Toc85288423)

[Pound 11](#_Toc85288424)

[Rock\_Tomb 11](#_Toc85288425)

[Stone\_Edge 12](#_Toc85288426)

[Swagger 12](#_Toc85288427)

[UML 13](#_Toc85288428)

[Пример выходных данных программы: 14](#_Toc85288429)

[Вывод 19](#_Toc85288430)

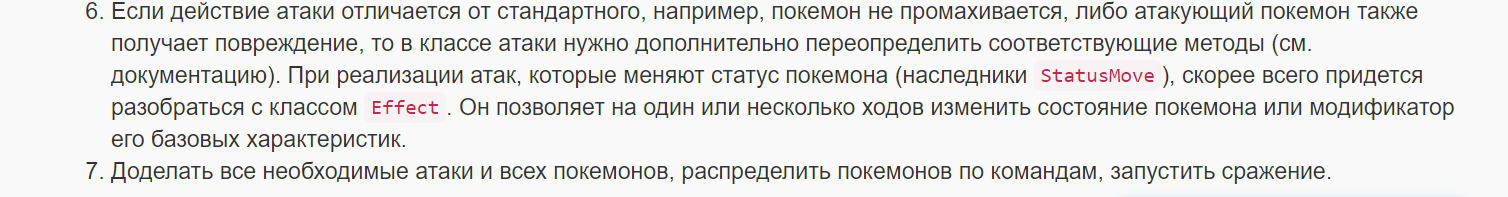
# Текст задания

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание



Изображение выглядит как текст

Автоматически созданное описание

# Покемоны и их атаки варианта:

Bouffalant:

Stone Edge

Leer

Rock Tomb

Facade

Cosmoem

Swagger

Double team

Munchlax

Blizzard

Body Slam

Double Team

Nigilego

Swagger

Double team

Aerial Ace

Charge Beam

Snorlax

Pound

Blizzard

Body Slam

Double Team

Solgaleo

Aerial Ace

Swagger

Double team

# Исходный код:

## Main

*import* ru.ifmo.se.pokemon.\*;  
*import* Pokemons.\*;  
  
*public class* Main {  
  
 *public static void* main(String[] args) {  
  
 Battle b = *new* Battle();  
  
 Munchlax p1 = *new* Munchlax("1", 22);  
 Bouffalant p2 = *new* Bouffalant("2", 30);  
 Snorlax p3 = *new* Snorlax("3", 38);  
 Cosmoem p4 = *new* Cosmoem("4", 43);  
 Solgaleo p5 = *new* Solgaleo("5", 53);  
 Nigilego p6 = *new* Nigilego("6", 61);  
  
 b.addAlly(p1);  
 b.addAlly(p2);  
 b.addAlly(p3);  
  
 b.addFoe(p4);  
 b.addFoe(p5);  
 b.addFoe(p6);  
  
 b.go();  
 }  
  
}

## Bouffalant

*package* Pokemons;  
  
*import* ru.ifmo.se.pokemon.\*;  
*import* Moves.\*;  
  
*public class* Bouffalant *extends* Pokemon {  
 *public* Bouffalant(String name, *int* level) {  
 *super*(name, level);  
 setStats(95, 110, 95, 30, 95, 55);  
 setType(Type.NORMAL);  
  
 addMove(*new* Stone\_Edge());  
 addMove(*new* Leer());  
 addMove(*new* Rock\_Tomb());  
 addMove(*new* Facade());  
 }  
}

## Cosmoem

*package* Pokemons;  
  
*import* ru.ifmo.se.pokemon.\*;  
*import* Moves.\*;  
  
*public class* Cosmoem *extends* Pokemon{  
 *public* Cosmoem (String name, *int* level) {  
 *super*(name, level);  
 setStats(43, 29, 131, 29, 131, 37);  
 setType(Type.PSYCHIC);  
 addMove(*new* Swagger());  
 addMove(*new* Double\_Team());  
 }  
  
}

## Munchlax

*package* Pokemons;  
  
*import* ru.ifmo.se.pokemon.\*;  
*import* Moves.\*;  
  
*public class* Munchlax *extends* Pokemon{  
 *public* Munchlax(String name, *int* level) {  
 *super*(name, level);  
 setStats(135, 85, 40, 40, 85, 5);  
 setType(Type.NORMAL);  
  
 addMove(*new* Blizzard());  
 addMove(*new* Body\_Slam());  
 addMove(*new* Double\_Team());  
 }  
  
}

## Nigilego

*package* Pokemons;  
  
*import* Moves.\*;  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Nigilego *extends* Pokemon {  
 *public* Nigilego(String name, *int* level) {  
 *super*(name, level);  
 setStats(109, 53, 47, 127, 131, 103);  
 setType(Type.ROCK, Type.POISON);  
  
 addMove(*new* Swagger());  
 addMove(*new* Double\_Team());  
 addMove((*new* Aerial\_Ace()));  
 addMove((*new* Charge\_Beam()));;  
 }  
}

## Snorlax

*package* Pokemons;  
  
*import* ru.ifmo.se.pokemon.\*;  
*import* Moves.\*;  
  
*public class* Snorlax *extends* Munchlax{  
 *public* Snorlax(String name, *int* level) {  
 *super*(name, level);  
 setStats(160, 110, 65, 65, 110, 30);  
 setType(Type.NORMAL);  
  
 addMove(*new* Pound());  
 }  
}

## Solgaleo

*package* Pokemons;  
  
*import* ru.ifmo.se.pokemon.\*;  
*import* Moves.\*;  
  
*public class* Solgaleo *extends* Cosmoem {  
 *public* Solgaleo(String name, *int* level) {  
 *super*(name, level);  
 setStats(137, 137, 107, 113, 89, 97);  
 setType(Type.PSYCHIC, Type.STEEL);  
  
 addMove(*new* Aerial\_Ace());  
 }  
}

## Aerial\_Ace

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Aerial\_Ace *extends* PhysicalMove {  
 *public* Aerial\_Ace() { *super*(Type.FLYING, 60, 0); } *// has 100% chance to hit  
  
 @Override  
 protected boolean* checkAccuracy(Pokemon pokemon, Pokemon pokemon1) {  
 *return true*;  
 }  
  
 *@Override  
 protected* String describe() { *return* "использует Aerial Ace"; }  
}

## Blizzard

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Blizzard *extends* SpecialMove {  
  
 *public* Blizzard() { *super* (Type.ICE, 110, 70); }  
  
 *@Override  
 protected void* applyOppEffects(Pokemon p) { *// 10% chance of freezing the target  
 if* (Math.random() <= 0.1) {  
 Effect.freeze(p);  
 }  
 }  
  
 *@Override  
 protected* String describe() { *return* "использует Blizzard"; }  
}

## Body\_Slam

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Body\_Slam *extends* PhysicalMove {  
  
 *public* Body\_Slam() { *super*(Type.NORMAL, 85, 100); }  
  
 *protected void* applyOppEffects(Pokemon p) { *// 30% chance of paralyzing the target  
 if* (Math.random() < 0.3) {  
 Effect.paralyze(p);  
 }  
 }  
  
 *@Override  
 protected* String describe() { *return* "использует Body Slam"; }  
}

## Charge\_Beam

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Charge\_Beam *extends* SpecialMove {  
 *public* Charge\_Beam() { *super*(Type.ELECTRIC, 50, 90); }  
  
 *@Override  
 protected void* applySelfEffects(Pokemon p) { *// has 70% chance of raising the user's attack by one stage  
 if* (Math.random() <= 0.7) {  
 p.setMod(Stat.ATTACK, 1);  
 }  
 }  
  
 *@Override  
 protected* String describe() { *return* "использует Charge Beam"; }  
}

## Double\_Team

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Double\_Team *extends* StatusMove{  
  
 *public* Double\_Team() { *super*(Type.NORMAL, 0, 0); }  
  
 *@Override  
 protected void* applySelfEffects(Pokemon p) { *// raises user's Evasiveness by one stage* p.setMod(Stat.EVASION, 1);  
 }  
  
 *@Override  
 protected* String describe () { *return* "использует Double Team"; }  
}

## Facade

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Facade *extends* PhysicalMove {  
  
 *public* Facade() {  
 *super*(Type.NORMAL, 70, 100);  
 }  
  
 *@Override  
 protected double* calcBaseDamage(Pokemon var1, Pokemon var2) { *// if the user is burned, poisoned or paralyzed deal damage with double power* Status c = var1.getCondition();  
 *int* d = 1; *// power multiplier  
 if* (c.equals(Status.BURN) || c.equals(Status.POISON) || c.equals(Status.PARALYZE)) {  
 d = 2;  
 }  
 *return* (0.4D \* (*double*) var1.getLevel() + 2.0D) \* *this*.power \* d / 150.0D;  
 }  
  
 *@Override  
 protected* String describe() { *return* "использует Facade"; }  
}

## Leer

*package* Moves;  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Leer *extends* StatusMove{  
 *public* Leer() {  
 *super*(Type.NORMAL, 0, 100);  
 }  
  
 *@Override  
 protected void* applyOppEffects(Pokemon p) { *// lowers the target's defence by 1 stage* p.setMod(Stat.DEFENSE, -1);  
 }  
  
 *@Override  
 protected* String describe() { *return* "использует Leer"; }  
}

## Pound

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Pound *extends* PhysicalMove {  
 *public* Pound() {  
 *super* (Type.NORMAL, 40, 100);  
 }  
  
 *@Override  
 protected* String describe() { *return* "использует Pound"; }  
}

## Rock\_Tomb

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Rock\_Tomb *extends* PhysicalMove {  
 *public* Rock\_Tomb() {  
 *super*(Type.ROCK, 60, 95);  
 }  
  
 *@Override  
 protected void* applyOppEffects(Pokemon p) { *// lower target's speed by 1 stage* p.setMod(Stat.SPEED, -1);  
 }  
  
 *@Override  
 protected* String describe() {  
 *return* "использует Rock Tomb";  
 }  
}

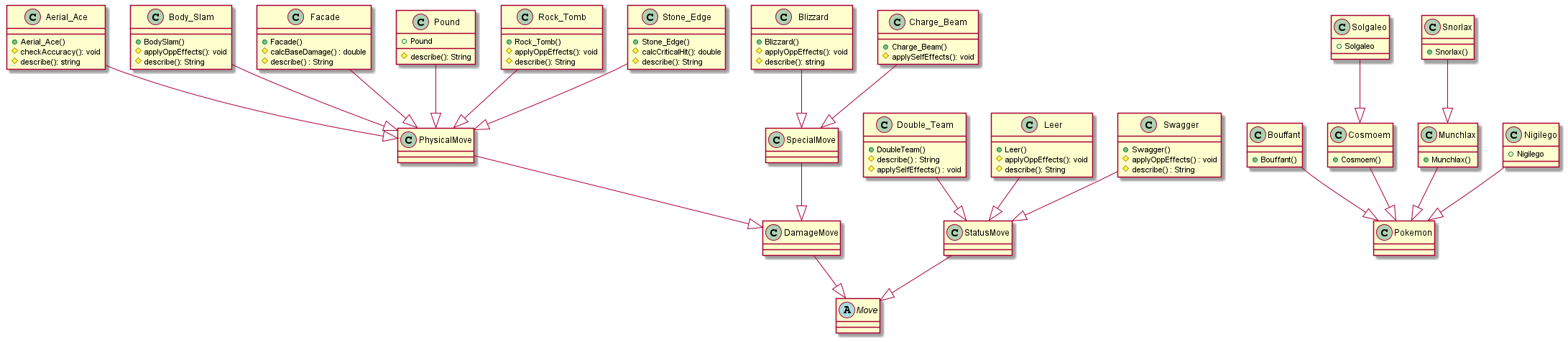
## Stone\_Edge

*package* Moves;  
  
*import* ru.ifmo.se.pokemon.\*;  
  
*public class* Blizzard *extends* SpecialMove {  
  
 *public* Blizzard() { *super* (Type.ICE, 110, 70); }  
  
 *@Override  
 protected void* applyOppEffects(Pokemon p) { *// 10% chance of freezing the target  
 if* (Math.random() <= 0.1) {  
 Effect.freeze(p);  
 }  
 }  
  
 *@Override  
 protected* String describe() { *return* "использует Blizzard"; }  
}

## Swagger

*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) { *// confuses the target and raises theirs attack by 2 stages* p.setMod(Stat.ATTACK, 2);  
 Effect.confuse(p);  
 }  
  
 *@Override  
 protected* String describe () { *return* "использует Swagger"; }  
}

# UML



# Пример выходных данных программы:

Munchlax 1 из команды синих вступает в бой!

Cosmoem 4 из команды желтых вступает в бой!

Cosmoem 4 борется с соперником.

Munchlax 1 теряет 19 здоровья.

Cosmoem 4 теряет 5 здоровья.

Munchlax 1 борется с соперником.

Cosmoem 4 теряет 7 здоровья.

Munchlax 1 теряет 2 здоровья.

Cosmoem 4 промахивается

Munchlax 1 промахивается

Cosmoem 4 промахивается

Munchlax 1 промахивается

Cosmoem 4 борется с соперником.

Munchlax 1 теряет 14 здоровья.

Cosmoem 4 теряет 4 здоровья.

Munchlax 1 использует Body Slam.

Cosmoem 4 теряет 7 здоровья.

Cosmoem 4 промахивается

Munchlax 1 использует Body Slam.

Cosmoem 4 теряет 11 здоровья.

Cosmoem 4 борется с соперником.

Munchlax 1 теряет 13 здоровья.

Cosmoem 4 теряет 3 здоровья.

Munchlax 1 промахивается

Cosmoem 4 борется с соперником.

Munchlax 1 теряет 19 здоровья.

Cosmoem 4 теряет 5 здоровья.

Munchlax 1 использует Body Slam.

Cosmoem 4 теряет 7 здоровья.

Cosmoem 4 парализован

Cosmoem 4 борется с соперником.

Критический удар!

Munchlax 1 теряет 30 здоровья.

Cosmoem 4 теряет 8 здоровья.

Munchlax 1 теряет сознание.

Bouffalant 2 из команды синих вступает в бой!

Cosmoem 4 использует Swagger.

Bouffalant 2 увеличивает атаку.

Bouffalant 2 использует Facade.

Cosmoem 4 теряет 25 здоровья.

Cosmoem 4 борется с соперником.

Критический удар!

Bouffalant 2 теряет 20 здоровья.

Cosmoem 4 теряет 5 здоровья.

Bouffalant 2 растерянно попадает по себе.

Bouffalant 2 теряет 10 здоровья.

Cosmoem 4 использует Swagger.

Bouffalant 2 увеличивает атаку.

Bouffalant 2 растерянно попадает по себе.

Bouffalant 2 теряет 15 здоровья.

Bouffalant 2 растерянно попадает по себе.

Bouffalant 2 теряет 16 здоровья.

Bouffalant 2 использует Rock Tomb.

Cosmoem 4 теряет 12 здоровья.

Cosmoem 4 уменьшает скорость.

Cosmoem 4 теряет сознание.

Solgaleo 5 из команды желтых вступает в бой!

Solgaleo 5 использует Swagger.

Bouffalant 2 увеличивает атаку.

Bouffalant 2 использует Rock Tomb.

Solgaleo 5 теряет 11 здоровья.

Solgaleo 5 уменьшает скорость.

Solgaleo 5 использует Swagger.

Bouffalant 2 увеличивает атаку.

Bouffalant 2 использует Facade.

Solgaleo 5 теряет 18 здоровья.

Solgaleo 5 промахивается

Bouffalant 2 использует Rock Tomb.

Solgaleo 5 теряет 12 здоровья.

Solgaleo 5 уменьшает скорость.

Solgaleo 5 использует Swagger.

Bouffalant 2 увеличивает атаку.

Bouffalant 2 растерянно попадает по себе.

Bouffalant 2 теряет 12 здоровья.

Solgaleo 5 использует Swagger.

Bouffalant 2 увеличивает атаку.

Bouffalant 2 использует Rock Tomb.

Solgaleo 5 теряет 11 здоровья.

Solgaleo 5 уменьшает скорость.

Bouffalant 2 использует Leer.

Solgaleo 5 уменьшает защиту.

Solgaleo 5 использует Aerial Ace.

Bouffalant 2 теряет 37 здоровья.

Bouffalant 2 теряет сознание.

Snorlax 3 из команды синих вступает в бой!

Snorlax 3 использует Body Slam.

Solgaleo 5 теряет 8 здоровья.

Solgaleo 5 борется с соперником.

Snorlax 3 теряет 34 здоровья.

Solgaleo 5 теряет 9 здоровья.

Snorlax 3 использует Body Slam.

Solgaleo 5 теряет 8 здоровья.

Solgaleo 5 борется с соперником.

Snorlax 3 теряет 25 здоровья.

Solgaleo 5 теряет 6 здоровья.

Snorlax 3 промахивается

Solgaleo 5 использует Aerial Ace.

Snorlax 3 теряет 39 здоровья.

Snorlax 3 использует Body Slam.

Solgaleo 5 теряет 6 здоровья.

Solgaleo 5 парализован

Solgaleo 5 использует Aerial Ace.

Snorlax 3 теряет 27 здоровья.

Snorlax 3 использует Pound.

Solgaleo 5 теряет 4 здоровья.

Solgaleo 5 использует Swagger.

Snorlax 3 увеличивает атаку.

Snorlax 3 использует Body Slam.

Solgaleo 5 теряет 11 здоровья.

Solgaleo 5 использует Aerial Ace.

Snorlax 3 теряет 45 здоровья.

Snorlax 3 использует Blizzard.

Solgaleo 5 теряет 7 здоровья.

Solgaleo 5 использует Aerial Ace.

Snorlax 3 теряет 40 здоровья.

Snorlax 3 теряет сознание.

В команде синих не осталось покемонов.

Команда желтых побеждает в этом бою!

# Вывод

Для выполнения этой лабораторной работы нужно было ознакомиться с основными принципами ООП, понятиями объектов, классов, наследования, инкапсуляции, научиться переопределять методы.