**Федеральное государственное автономное образовательное**

**учреждение высшего образования**

**«Национальный исследовательский университет ИТМО»**

Факультет программной инженерии и компьютерной техники

Программирование

**Лабораторная работа №2**

**Студент**: Фам Данг Чунг Нгиа

**Группа:** P3121

**Преподаватель**: Лисицина Василиса Васильевна

САНКТ-ПЕТЕРБУРГ

2023

Оглавление

[1. Текст задания. 3](#_Toc149088033)

[2. Диаграмма классов реализованной объектной модели 5](#_Toc149088034)

[3. Исходный код программы 5](#_Toc149088035)

[4. Результат работы программы 7](#_Toc149088036)

[5. Выводы по работе 9](#_Toc149088037)

# 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.

Базовые классы и симулятор сражения находятся в [jar-архиве](https://se.ifmo.ru/documents/10180/660917/Pokemon.jar/a7ce60af-6ee6-47d0-a95e-e5ed9a697bd2) (обновлен 9.10.2018, исправлен баг с добавлением атак и кодировкой). Документация в формате javadoc - [здесь](https://se.ifmo.ru/~tony/doc/).

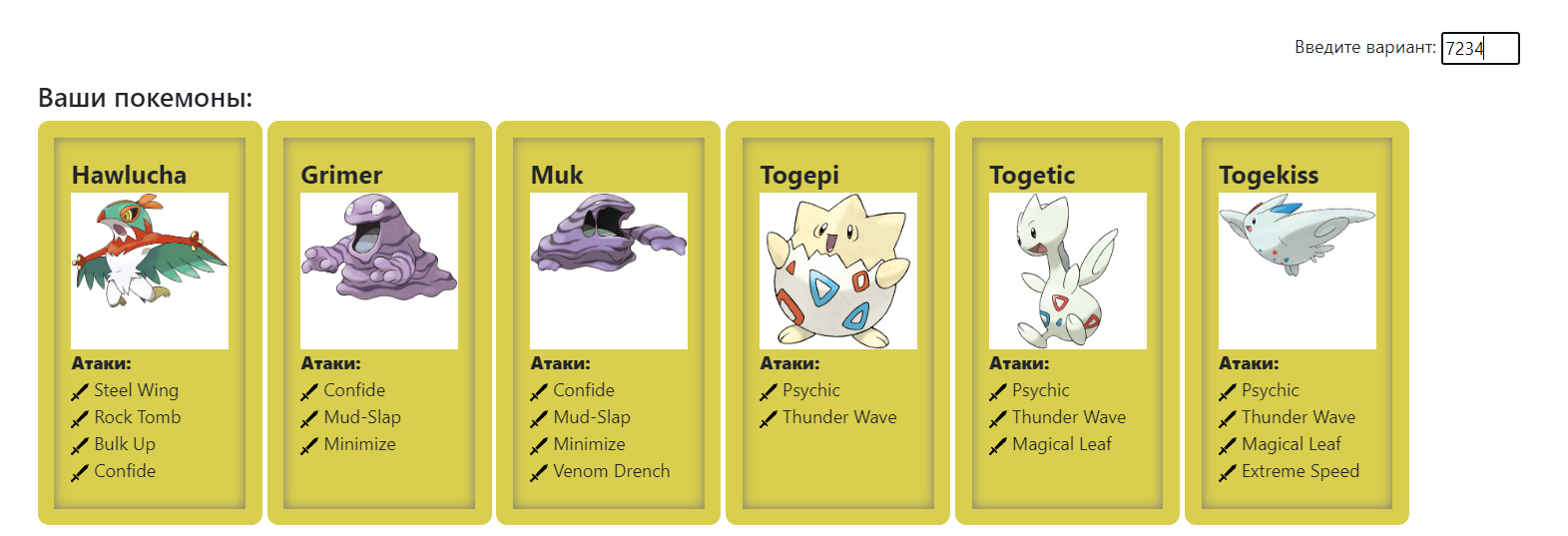
Информацию о покемонах, цепочках эволюции и атаках можно найти на сайтах [http://poke-universe.ru](http://poke-universe.ru/), [http://pokemondb.net](http://pokemondb.net/),<http://veekun.com/dex/pokemon>

#### **Комментарии**

Цель работы: на простом примере разобраться с основными концепциями ООП и научиться использовать их в программах.

Что надо сделать (краткое описание)

1. Ознакомиться с [документацией](https://se.ifmo.ru/~tony/doc/), обращая особое внимание на классы Pokemon и Move. При дальнейшем выполнении лабораторной работы читать документацию еще несколько раз.
2. Скачать файл Pokemon.jar. Его необходимо будет использовать как для компиляции, так и для запуска программы. Распаковывать его не надо! Нужно научиться подключать внешние jar-файлы к своей программе.
3. Написать минимально работающую программу и посмотреть как она работает.
4. Battle b = new Battle();
5. Pokemon p1 = new Pokemon("Чужой", 1);
6. Pokemon p2 = new Pokemon("Хищник", 1);
7. b.addAlly(p1);
8. b.addFoe(p2);
9. b.go();
10. Создать один из классов покемонов для своего варианта. Класс должен наследоваться от базового класса Pokemon. В конструкторе нужно будет задать типы покемона и его базовые характеристики. После этого попробуйте добавить покемона в сражение.
11. Создать один из классов атак для своего варианта (лучше всего начать с физической или специальной атаки). Класс должен наследоваться от класса PhysicalMove или SpecialMove. В конструкторе нужно будет задать тип атаки, ее силу и точность. После этого добавить атаку покемону и проверить ее действие в сражении. Не забудьте переопределить метод describe, чтобы выводилось нужное сообщение.
12. Если действие атаки отличается от стандартного, например, покемон не промахивается, либо атакующий покемон также получает повреждение, то в классе атаки нужно дополнительно переопределить соответствующие методы (см. документацию). При реализации атак, которые меняют статус покемона (наследники StatusMove), скорее всего придется разобраться с классом Effect. Он позволяет на один или несколько ходов изменить состояние покемона или модификатор его базовых характеристик.
13. Доделать все необходимые атаки и всех покемонов, распределить покемонов по командам, запустить сражение.

Мой вариант: **7234**

# Диаграмма классов реализованной объектной модели

# Исходный код программы

Мой репозиторий:

<https://github.com/nghiaphamhb/Programming/tree/main/%D0%9B%D0%B0%D0%B1%202/src>

1. Главный класс (main)

import ru.ifmo.se.pokemon.\*;  
import Pokemons.\*;  
public class Battle {  
 public static void main(String[] args){  
 ru.ifmo.se.pokemon.Battle b = new ru.ifmo.se.pokemon.Battle();  
 Pokemon p1 = new Grimer("A", 10);  
 Pokemon p2 = new Hawlucha("B",10 );  
 Pokemon p3 = new Muk("C", 10);  
 Pokemon p4 = new Togekiss("D", 1);  
 Pokemon p5 = new Togepi("E", 2);  
 Pokemon p6 = new Togetic("B", 1);  
 b.addAlly(p1);  
 b.addAlly(p2);  
 b.addAlly(p3);  
 b.addFoe(p4);  
 b.addFoe(p5);  
 b.addFoe(p6);  
 b.go();  
 }  
}

1. Всякий класс наследующийся от класса **Pokemon**

package Pokemons;  
import Moves.SpecialMoves.MudSlap;  
import Moves.StatusMoves.Confide;  
import Moves.StatusMoves.Minimize;  
import ru.ifmo.se.pokemon.\*;  
public class Grimer extends Pokemon{  
 public Grimer(String name, int level){  
 super(name, level);  
 setType(Type.*POISON*);  
 setStats(80, 80, 50,40,50,25);  
 setMove(new Confide(), new MudSlap(), new Minimize());  
 }  
}

1. Всякий класс наследующийся от класса **Move**

package Moves.PhysicalMoves;  
import ru.ifmo.se.pokemon.\*;  
public class SteelWing extends PhysicalMove {  
 public SteelWing(){  
 super(Type.*STEEL*, 70, 90);  
 }  
 @Override  
 protected void applySelfEffects(Pokemon pokemon) {  
 if(Math.*random*()<=0.1){  
 pokemon.setMod(Stat.*DEFENSE*, 1);  
 }  
 }  
 protected String describe(){  
 return "использует способность Steel Wing";  
 }  
}

# Результат работы программы

**Grimer A** from the team **Blue** enters the battle!

**Togekiss D** from the team **Red** enters the battle!

**Grimer A** использует способность **Mud Slap**.

**Togekiss D** loses 1 hit points.

**Togekiss D** isn't affected by **GROUND**

**Togekiss D** использует способность **Psychic**.

**Grimer A** loses 8 hit points.

**Togekiss D** использует способность **Extreme Speed**.

**Grimer A** loses 3 hit points.

**Grimer A** misses

**Togekiss D** использует способность **Extreme Speed**.

**Grimer A** loses 4 hit points.

**Grimer A** misses

**Grimer A** misses

**Togekiss D** использует способность **Psychic**.

**Grimer A** loses 6 hit points.

**Togekiss D** использует способность **Extreme Speed**.

**Grimer A** loses 4 hit points.

**Grimer A** использует способность **Mud Slap**.

**Togekiss D** loses 1 hit points.

**Togekiss D** isn't affected by **GROUND**

Grimer A использует способность Mud Slap.

Togekiss D loses 1 hit points.

Togekiss D isn't affected by GROUND

Togekiss D использует способность Psychic.

Grimer A loses 8 hit points.

Grimer A misses

Togekiss D использует способность Psychic.

Grimer A loses 9 hit points.

Grimer A faints.

Hawlucha B from the team Blue enters the battle!

Hawlucha B использует способность Steel Wing.

Togekiss D loses 9 hit points.

Togekiss D использует способность Thunder Wave.

Hawlucha B is paralyzed

Hawlucha B использует способность Rock Tomb.

Critical hit!

Togekiss D loses 76 hit points.

Togekiss D decreases speed.

Togekiss D faints.

Togepi E from the team Red enters the battle!

Hawlucha B misses

Togepi E использует способность Psychic.

Hawlucha B loses 7 hit points.

Hawlucha B misses

Togepi E использует способность Thunder Wave.

Hawlucha B misses

Togepi E использует способность Thunder Wave.

Hawlucha B misses

Togepi E использует способность Psychic.

Hawlucha B loses 9 hit points.

Hawlucha B misses

Togepi E использует способность Thunder Wave.

Hawlucha B misses

Togepi E использует способность Psychic.

Hawlucha B loses 5 hit points.

Hawlucha B использует способность Rock Tomb.

Critical hit!

Togepi E loses 36 hit points.

Togepi E decreases speed.

Togepi E faints.

Togetic B from the team Red enters the battle!

Hawlucha B misses

Togetic B использует способность Thunder Wave.

Hawlucha B misses

Togetic B использует способность Magical Leaf.

Hawlucha B loses 2 hit points.

Hawlucha B misses

Togetic B использует способность Thunder Wave.

Hawlucha B использует способность Steel Wing.

Critical hit!

Togetic B loses 21 hit points.

Togetic B faints.

**Team Red loses its last Pokemon.**

**The team Blue wins the battle!**

# Выводы по работе

Я научился тем, как разобраться с основными концепциями ООП и научиться использовать их в программах. Я использовал класс симуляции боя Battle, создал 2 команды покемонов (каждый покемон должен иметь имя) и запуститл бой.