

**Федеральное государственное автономное образовательное учреждение высшего  
образования «Национальный исследовательский университет ИТМО»**

**Факультет программной инженерии и компьютерной техники**

**Лабораторная работа №2**

По программированию

“Основы ООП”

Вариант №2624

Выполнил:

**Ступин Тимур Русланович**

Группа № Р3108

Поток № 2.6

Проверила:

**Харитоновна Анастасия  
Евгеньевна**

Санкт-Петербург 2023

## Содержание

Задание .....	3
Диаграммы классов объектной модели .....	4
Исходный код программы.....	4
Результат работы программы.....	4
Вывод .....	6

## 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.

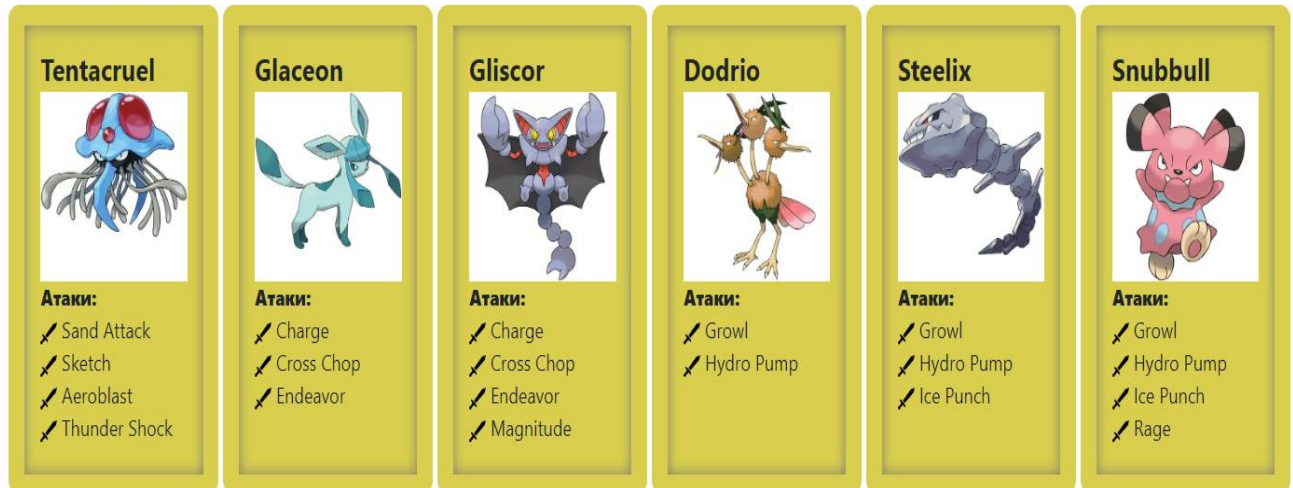
Атака должна иметь стандартные тип, силу (power) и точность (accuracy). Должны быть реализованы стандартные эффекты атаки. Назначить каждому виду покемонов атаки в соответствии с вариантом. Уровень покемона выбирается минимально необходимым для всех реализованных атак.

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>

Ваши покемоны:



## Диаграммы классов объектной модели

[Ссылка на pdf файл с UML диаграммой проекта.](#)

## Исходный код программы

[Ссылка на репозиторий GitHub.](#)

## Результат работы программы

```
[s409642@helios ~/prog_lab2]$ java -cp ../Pokemon.jar:main.jar
lab2.Main
Picked up _JAVA_OPTIONS: -XX:MaxHeapSize=1G -
XX:MaxMetaspaceSize=128m
Tentacruel Tent из команды зеленых вступает в бой!
Glaceon Glac из команды синих вступает в бой!
Tentacruel Tent промахивается

Glaceon Glac использует атаку Endeavor.
Tentacruel Tent теряет 4 здоровья.

Tentacruel Tent использует специальную атаку Aeroblast.
Glaceon Glac теряет 6 здоровья.

Tentacruel Tent теряет 2 здоровья.
Glaceon Glac использует атаку Endeavor.
Tentacruel Tent теряет 4 здоровья.

Tentacruel Tent выполняет Sand Attack.

Glaceon Glac использует атаку Cross Chop.
Критический удар!
Tentacruel Tent теряет 4 здоровья.
```

Tentacruel Tent теряет сознание.  
Gliscor Glis из команды зеленых вступает в бой!  
Gliscor Glis промахивается

Gliscor Glis теряет 6 здоровья.  
Glaceon Glac использует атаку Endeavor.  
Gliscor Glis теряет 3 здоровья.

Gliscor Glis использует атаку Magnitude.  
Glaceon Glac теряет 6 здоровья.

Glaceon Glac промахивается

Gliscor Glis использует атаку Cross Chop.  
Glaceon Glac теряет 12 здоровья.  
Glaceon Glac теряет сознание.  
Dodrio Dodr из команды синих вступает в бой!  
Dodrio Dodr выполняет Growl.

Gliscor Glis промахивается

Dodrio Dodr использует специальную атаку Hydro Pump.  
Gliscor Glis теряет 11 здоровья.  
Gliscor Glis теряет сознание.  
Steelix Stee из команды зеленых вступает в бой!  
Dodrio Dodr выполняет Growl.

Steelix Stee использует специальную атаку Hydro Pump.  
Dodrio Dodr теряет 6 здоровья.

Dodrio Dodr использует специальную атаку Hydro Pump.  
Steelix Stee теряет 10 здоровья.

Steelix Stee выполняет Growl.

Dodrio Dodr использует специальную атаку Hydro Pump.  
Steelix Stee теряет 9 здоровья.  
Steelix Stee теряет сознание.  
В команде зеленых не осталось покемонов.  
Команда синих побеждает в этом бою!

## **Вывод**

В процессе работы я изучил основные концепции ООП и научился использовать их при программировании на языке Java. Я научился использовать сторонний jar файл в своем проекте, импортировать и наследовать из него классы. Также я научился создавать пакеты в java, компилировать и запускать их с использованием сторонних jar файлов средствами терминала Linux, а также создавать на их основе UML диаграммы, используя средства интегрированной среды разработки IntelliJ IDEA.