

**Федеральное государственное автономное
образовательное учреждение
высшего образования
«Национальный исследовательский университет
ИТМО»
Факультет Программной Инженерии и Компьютерной
Техники**



**Вариант №1500235
Лабораторная работа №2
по дисциплине
Программирование**

**Выполнил Студент группы Р3115
Владимир Мацюк
Преподаватель:
Сорокин Роман Борисович**

**г. Санкт-Петербург
2022г.**

Содержание

1 Текст задания	1
2 Комментарии	1
3 Покемоны	2
4 Диаграмма классов	3
5 Ссылка на github	3
6 Результат работы программы	3
7 Вывод	4

1 Текст задания

1. 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)
2. 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.
3. 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.
4. Используя класс симуляции боя `Battle`, создать 2 команды покемонов (каждый покемон должен иметь имя) и запустить бой.
5. 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](#).
6. Информацию о покемонах, цепочках эволюции и атаках можно найти на сайтах <http://poke-universe.ru>, <http://pokemondb.net>, <http://veekun.com/dex/pokemon>

2 Комментарии

1. Ознакомиться с документацией, 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.

```

1 Battle b = new Battle();
2 Pokemon p1 = new Pokemon("Чужой", 1);
3 Pokemon p2 = new Pokemon("Хищник", 1);
4 b.addAlly(p1);
5 b.addFoe(p2);
6 b.go();

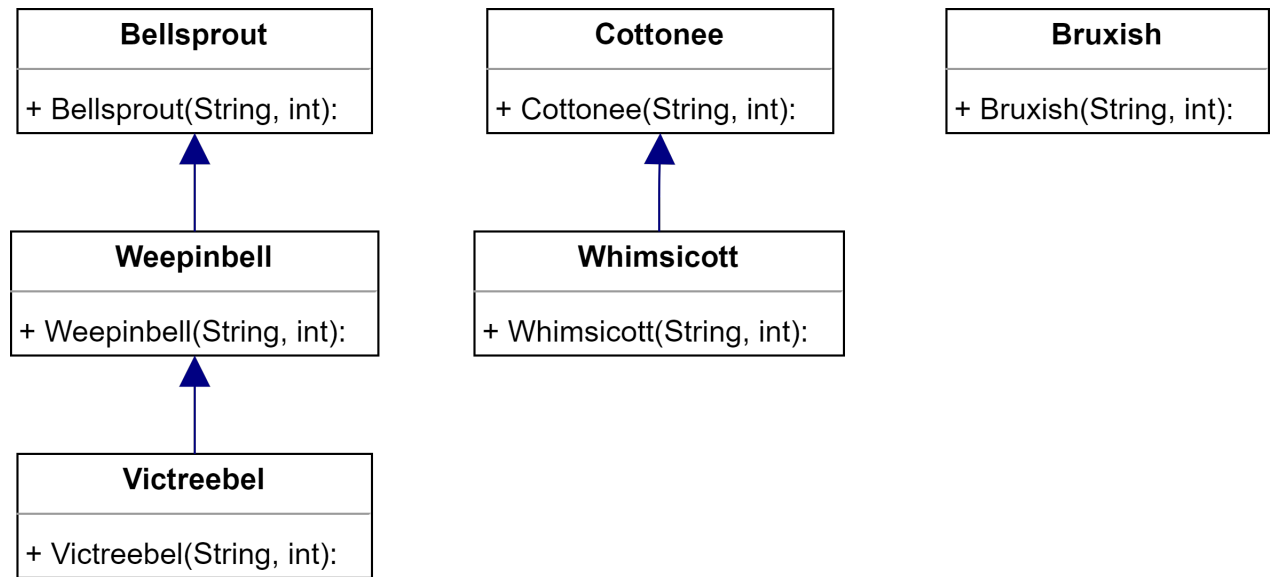
```

4. 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.
5. 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.
6. 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.
7. Доделать все необходимые атаки и всех покемонов, распределить покемонов по командам, запустить сражение.

3 Покемоны

Bruxish  Атаки: ✓ Slash ✓ Dark Pulse ✓ Hone Claws ✓ Will-O-Wisp	Cottonee  Атаки: ✓ Energy Ball ✓ Rest ✓ Dream Eater	Whimsicott  Атаки: ✓ Energy Ball ✓ Rest ✓ Dream Eater ✓ Hurricane	Bellsprout  Атаки: ✓ Sludge Bomb ✓ Sleep Powder	Weepinbell  Атаки: ✓ Sludge Bomb ✓ Sleep Powder ✓ Acid	Victreebel  Атаки: ✓ Sludge Bomb ✓ Sleep Powder ✓ Acid ✓ Energy Ball
--	---	--	---	--	---

4 Диаграмма классов



5 Ссылка на github

<https://github.com/Wgmlgz/itmo/tree/main/prog/lab2>

6 Результат работы программы

out.txt

```
1 Bellsprout Geralt of Rivia from the team Purple enters the battle!
2 Victreebel Triss Merigold from the team Black enters the battle!
3 Bellsprout Geralt of Rivia misses
4
5 Victreebel Triss Merigold uses EnergyBall.
6 Bellsprout Geralt of Rivia loses 2 hit points.
7
8 Bellsprout Geralt of Rivia misses
9
10 Victreebel Triss Merigold misses
11
12 Bellsprout Geralt of Rivia misses
13
14 Victreebel Triss Merigold uses Acid.
15 Bellsprout Geralt of Rivia loses 5 hit points.
16
17 Bellsprout Geralt of Rivia misses
18
19 Victreebel Triss Merigold misses
20
21 Bellsprout Geralt of Rivia misses
22
23 Victreebel Triss Merigold misses
24
25 Bellsprout Geralt of Rivia misses
26
27 Victreebel Triss Merigold uses Acid.
```

```
28 Bellsprout Geralt of Rivia loses 8 hit points.
29 Bellsprout Geralt of Rivia faints.
30 Bruxish Ciri from the team Purple enters the battle!
31 Bruxish Ciri uses Slash .
32 Victreebel Triss Merigold loses 14 hit points.
33 Victreebel Triss Merigold faints.
34 Weepinbell Dandelion from the team Black enters the battle!
35 Bruxish Ciri misses
36
37 Weepinbell Dandelion uses Acid.
38 Bruxish Ciri loses 7 hit points.
39
40 Bruxish Ciri misses
41
42 Weepinbell Dandelion misses
43
44 Bruxish Ciri misses
45
46 Weepinbell Dandelion uses Acid.
47 Bruxish Ciri loses 7 hit points.
48 Bruxish Ciri faints.
49 Cottonee Yennefer of Vengerberg from the team Purple enters the battle!
50 Cottonee Yennefer of Vengerberg uses DreamEater.
51 Weepinbell Dandelion loses 13 hit points.
52 Weepinbell Dandelion is sleeping
53 Weepinbell Dandelion faints.
54 Whimsicott Vesemir from the team Black enters the battle!
55 Whimsicott Vesemir uses Hurricane.
56 Cottonee Yennefer of Vengerberg loses 14 hit points.
57 Cottonee Yennefer of Vengerberg faints.
58 Team Purple loses its last Pokemon.
59 The team Black wins the battle!
```

7 Вывод

Во время выполнения работы я ознакомился с ООП на языке java. Научился разрабатывать архитектуру проекта и подключать jar архивы к проекту.