Санкт-Петербургский Национальный Исследовательский Университет ИТМО

Факультет Программной Инженерии и Компьютерной Техники



Вариант №937

Лабораторная работа №2

По дисциплине

Программирование

Выполнил студент группы P3130:

Оладое Майкл Блессинг

Преподаватель:

Гаврилов Антон Валерьевич

Кустарев Иван Павлович

Санкт-Петербург 2024 г.

1. **Текст задания**

Write your own Pokémon classes based on Pokémon class for all given Pokémons. Each Pokémon kind should have one or two types and standard base stats: HP, attack, defense,

special attack, special defense and speed.

Pokémon classes should be inherited according to Pokémon evolution chains.

Write your own Move classes based on PhysicalMove, SpecialMove and StatusMove classes for all give moves. Each move should have standard type, power and accuracy and implement standard move effects. Assign moves to Pokémons according to given task. Pokémon level should be set to minimal one required to learn all given moves.

Use the simulation class Battle to create two Pokémon teams (each Pokémon should have a name) and start the battle.

Base classes, battle simulator and utility classes are packed in jar archive. Documentation in javadoc format is in the zip file.

All information about pokemon and move stats, evolution chains and so on you can find on <http://pokemondb.net>, <http://veekun.com/dex/pokemon>



1. **Исходный код программы.**

Репозиторий: <https://github.com/Jimike110/ITMO/tree/4b5b0c70e4fd7c95ee92dc35e81240f9c6b4ea0a/first_year/first_semester/programming/labs/lab2/src>

1. **Диаграмма классов реализованной объектной модели.**

A screenshot of a computer

Description automatically generated

1. **Результат работы программы:**

См. в репозитории: <https://github.com/Jimike110/ITMO/blob/4b5b0c70e4fd7c95ee92dc35e81240f9c6b4ea0a/first_year/first_semester/programming/labs/lab2/logs/result.log>

1. **Вывод**

В процессе выполнения данного задания я научился работать с jar-файлами в своей программе, используя их как при написании кода, так и при компиляции. Я также работал с инструментом для создания UML-диаграмм, Code2Colors. Задание помогло мне глубже понять принципы объектно-ориентированного программирования (ООП), такие как наследование и модификатор @Override.

