

**МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ**

**ОТЧЕТ
по лабораторной работе №2
по дисциплине «Объектно-ориентированное программирование»
Тема: «Интерфейсы классов; взаимодействие классов; перегрузка
операций»**

Студентка гр. 8304

Николаева М.А.

Преподаватель

Размочаева Н.В.

Санкт-Петербург
2020

Цель работы.

Научиться создавать классы, конструкторы классов и методы. Освоить наследование классов. Разработать и реализовать классы ландшафтов и набор классов нейтральных объектов, класс базы.

Постановка задачи.

Разработать и реализовать набор классов:

- Класс базы
- Набор классов ландшафта карты
- Набор классов нейтральных объектов поля

Класс базы должен отвечать за создание юнитов, а также учитывать юнитов, относящихся к текущей базе. Основные требования к классу база:

- База должна размещаться на поле
- Методы для создания юнитов
- Учет юнитов, и реакция на их уничтожение и создание
- База должна обладать характеристиками такими, как здоровье, максимальное количество юнитов, которые могут быть одновременно созданы на базе, и т.д.

Набор классов ландшафта определяют вид поля. Основные требования к классам ландшафта:

Должно быть создано минимум 3 типа ландшафта

- Все классы ландшафта должны иметь как минимум один интерфейс
- Ландшафт должен влиять на юнитов (например, возможно пройти по клетке с определенным ландшафтом или запрет для атаки определенного типа юнитов)
- На каждой клетке поля должен быть определенный тип ландшафта

Набор классов нейтральных объектов представляют объекты, располагаемые на поле и с которыми могут взаимодействие юнитов. Основные требования к классам нейтральных объектов поля:

- Создано не менее 4 типов нейтральных объектов
- Взаимодействие юнитов с нейтральными объектами, должно быть реализовано в виде перегрузки операций
- Классы нейтральных объектов должны иметь как минимум один общий интерфейс

Ход выполнения работы.

- 1) Был реализован класс базы. Класс базы является наследником класса Unit, соответственно имеет возможность размещения на поле и обладает всеми требуемыми характеристиками (здоровье и тп). Создано поле, для учета количества юнитов у базы. Данный класс содержит методы для создания трех типов юнитов (infantry-, flying-, standingUnit). Для случайного выбора вида юнита для каждого типа у базы создан приватный метод получения случайного числа.
- 2) Был реализован абстрактный класс ландшафт. Он наследуется от интерфейса (iDrawable). От класса ландшафта наследуются классы для четырех типов ландшафта, которые отличаются видом воздействия на юнитов, при этом один и тот же ландшафт может по-разному влиять на разные типы юнитов. Использован паттерн «Прoxy». Клетка игрового поля содержит указатель на ландшафт, размещенный на клетке.
- 3) Реализован абстрактных класс нейтрального объекта, наследуемый от интерфейса (iDrawable). От него наследуются четыре типа объектов, которые соответственно имеют разные виды воздействия на юнитов. При этом для объекта, повышающего уровень брони, влияние на различные виды юнитов также различно. Воздействия на юнитов реализовано с помощью перегрузки оператора круглые скобки.
- 4) Для реализации воздействия объектов написаны классы стратегий. По одному типу для всех видов нейтральных объектов, кроме воздействующего на броню (для него дополнительно реализованы три вида стратегий, так как юниты могут иметь различные виды брони).

- 5) В main созданы функции для тестирования программы (тестирование возможностей базы, влияния ландшафта на юнитов, влияния нейтральных объектов на юнитов).

Выводы.

В ходе выполнения работы были разработаны и реализованы класс базы, набор классов ландшафта, набор классов нейтральных объектов. Были выполнены все основные требования к данной лабораторной работе. Методы принимают параметры оптимальным образом. Для удобного восприятия классов создана UML диаграмма.