

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

**ОТЧЕТ
по лабораторной работе №1
по дисциплине «Объектно-ориентированное программирование»
Тема: «Создание классов, конструкторов класса, методов класса,
наследование»**

Студент гр. 8304

Сани З. Б

Преподаватель

Размочаева Н.В.

Санкт-Петербург
2020

Цель работы.

Научиться создавать классы и их конструкторы, реализовать методы классов и познакомиться с наследованием классов.

Постановка задачи.

Разработать и реализовать набор классов:

- Класс игрового поля
- Набор классов юнитов

Игровое поле является контейнером для объектов представляющим прямоугольную сетку. Основные требования к классу игрового поля:

- Создание поля произвольного размера
- Контроль максимального количества объектов на поле
- Возможность добавления и удаления объектов на поле
- Возможность копирования поля (включая объекты на нем)
- Для хранения запрещается использовать контейнеры из STL
- *Создан итератор для поля*

Юнит является объектом, размещаемым на поле боя. Один юнит представляет собой отряд. Основные требования к классам юнитов:

- Все юниты должны иметь как минимум один общий интерфейс
- Реализованы 3 типа юнитов (например, пехота, лучники, конница)
- Реализованы 2 вида юнитов для каждого типа (например, для пехоты могут быть созданы мечники и копейщики)
- Юниты имеют характеристики, отражающие их основные атрибуты, такие как здоровье, броня, атака.
- Юнит имеет возможность перемещаться по карте

- Для создания юнитов используются паттерны “Абстрактный метод” / “Абстрактная фабрика”
- Для атрибутов юнитов созданы свои классы. Создавать их требуется, если это не противоречит логике.
- Созданы конструкторы копирования и перемещения
- Все методы принимают параметры оптимальным образом (то есть, отсутствует лишнее копирование объектов)

Ход выполнения.

- 1) Создан абстрактный базовый класс объекта, реализованы три подкласса и для каждого подкласса создано по 2 подкласса. Иерархия классов показана на рис. 1

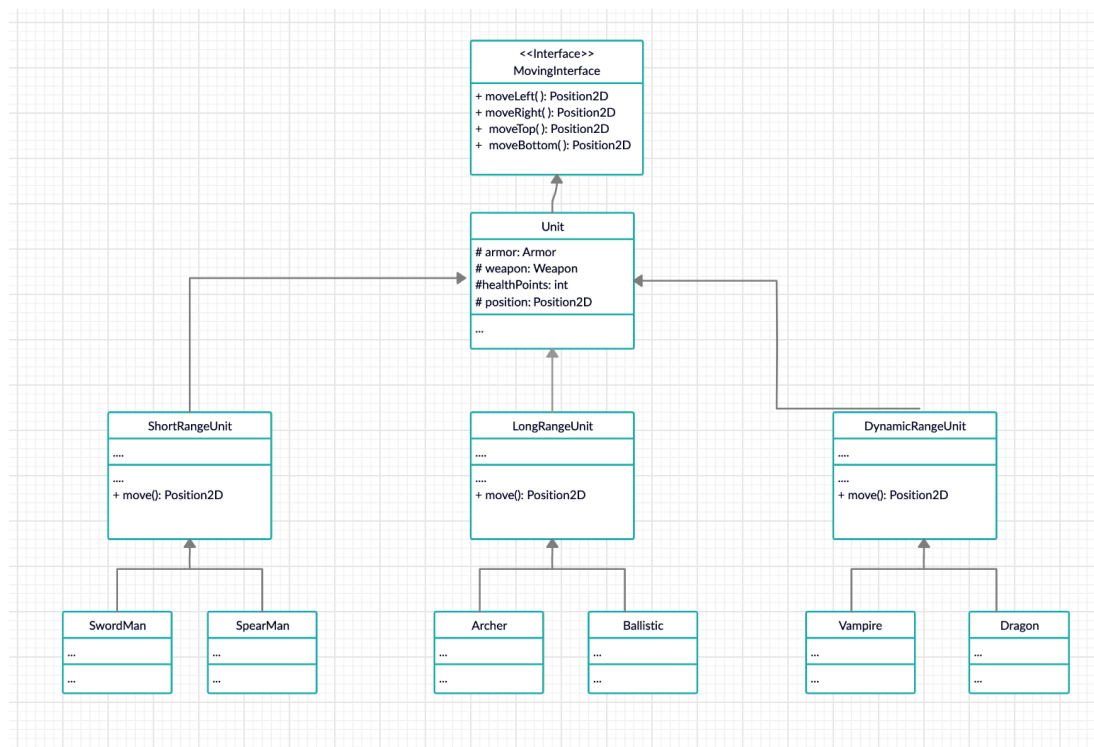


Рисунок 1 – Иерархия классов Юнит

- 2) Были созданы иерархии классов атрибутов, которые включает в себя юнит. Для клонирования был использован паттерн «Prototype».

Armor – abstract class

armorType

resistance

1. No Armor
2. Light Armor
3. Medium Armor
4. Heavy Armor

Weapon – abstract class

weaponType

damage

radius

distance

- 3) Для создания юнитов и атрибутов были использованы паттерны, Factory pattern and Builder pattern.
- 4) были созданы классы ячейки поля, игрового поля и итератора. Контейнеры STL не использовались. В конструкторе полей можно задать произвольный размер. Поле имеет счетчик юнит, который используется для управления максимальным количеством юнитов в игровом поле, и юнит могут быть добавлены или удалены с поля. Конструкторы копирования и перемещения были определены для поля и всех зависимых классов.
- 5) все методы принимают аргументы оптимальным образом.

Выводы.

В ходе работы были изучены и реализованы способы создания классов и их конструкторов, а также изучены и унаследованы абстрактные классы

.Весь код написан с использованием «умных» указателей.