

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

ОТЧЕТ
по лабораторной работе №6
по дисциплине «Объектно-ориентированное программирование»
Тема: Шаблонные классы

Студент гр. 8304

Бутко А.М.

Преподаватель

Размочаева Н.В.

Санкт-Петербург

2020

Цель работы.

Разработать и реализовать набор классов правил игры.

Выполнение работы

| Требование к работе | Выполнение требования | |
|--|-----------------------|---|
| Правила игры должны определять начальное состояние игры. | + | Правила определяют размеры поля, расположение баз, лимит количества хранимых на базе юнитов, количество игроков. |
| Правила игры должны определять условия выигрыша игроков. | + | Первое правило (RuleOne.h/.cpp) определяет победу игрока над остальными игроками, если базы противников разрушены. Создавать и хранить на базе можно только 5 юнитов. Второе правило (RuleTwo.h/.cpp) определяет поражение игрока, если у него разрушена база или счетчик умерших на поле юнитов равен 10. Создавать и хранить на базе можно только 3 юнита. |
| Правила игры должны определять очередность ходов игрока. | + | При двух (трех) игроках ход передается поочередно. Ход передается в случае создания игроком юнита, выставления юнита на поле, передвижения юнитом, атаки юнитом или если игрок решил пропустить ход («/skip»). В случае запроса информации о базе, включения логгирования или его выключения ход не передается. |
| Должна быть возможность начать новую игру. | + | Игрок может ввести команду «/new_game» для того чтобы в конфигурации (правило - количество игроков) начать новую игру. |
| Должно быть реализовано минимум 2 правила игры | + | Реализованы два правила (RuleOne и RuleTwo), которые описаны выше. |
| <i>* Класс игры в шаблоне поддерживает кол-во игроков. И для определенного кол-ва должен быть специализирован отдельно</i> | + | При выборе конфигурации выбирается шаблон <правило, количество игроков>, но для различного количества игроков класс игры реализован по-разному. Класс игры расположен в файле GameClass.h (бывший Facade.h). |

| | | |
|---|---|--|
| <i>* Передача хода реализована при помощи паттерна «Состояние».</i> | + | Для каждого игрока определяется состояния игры (от чьей игровой стороны играет игрок), файлы StateBright.h/.cpp, StateDark.h/.cpp, StateUndead.h/.cpp. |
| <i>* Класс игры единственный и создается партерном «Синглтон».</i> | + | Класс игры GameClass, в котором и создаются все элементы игры, а так же из которого ведется все управление игрой. |

Выводы.

Были реализованы основные требования к данной лабораторной работе, а так же изучены паттерны «Синглтон» и «Состояние» а так же шаблоны в классах.