

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

ОТЧЕТ
по лабораторной работе №5
по дисциплине «Объектно-ориентированное программирование»
Тема: Сериализация состояния программы

Студентка гр. 8382

Рочева А.К.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2020

Цель работы.

Реализовать возможность записи состояния программы в файл и возможность считывания состояния программы из файла.

Ход выполнения.

Был создан класс Memento (в Memento.hpp), хранящий состояние программы. Так же для удобства были созданы отдельные структуры для хранения состояния игры, базы и юнита (GameState, BaseState, UnitState, все определены в файле Memento.hpp).

Для сохранения игры в класс Memento передается указатель на игровое поле и указатель на map, содержащий игроков и их базы. В методе saveGame характеристики поля, баз и юнитов, относящихся к этим базам сохраняются в выше описанные структуры. Затем все сохраняется в файл (если пользователь хочет сохранить состояние только в классе, он вводит вместо пути в файл знак '-').

В документе структура состояния программы описана так:

```
fieldWidth fieldHeight
LandName x y
...
LandName x y
numObject
ObjName x y
...
ObjName x y
numBases
GamerName
xBase yBase
baseHealth
numUNits
UnitName x y attack defense health
...
UnitName x y attack defense health
```

Загрузка состояния из документа происходит в методе `loadGameFromDoc()`.

Состояние помимо документов хранятся в классе `Manager`. Пользователь в любой момент может сохранить игру или загрузить старое состояние.

Выводы.

В ходе выполнения работы был разработан класс `Memento` для хранения состояния игры. Реализовано сохранения состояния в файл и считывание его из файла.