

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

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

Студентка гр. 8381

Звегинцева Е.Н.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2020

## **Цель работы.**

Реализация сохранения и загрузки программы.

## **Задание.**

Основные требования:

- Возможность записать состояние программы в файл
- Возможность считать состояние программы из файла
- Загрузка и сохранение должно выполняться в любой момент

программы

- Взаимодействие с файлами должны быть по идиоме RAII
- Сохранение и загрузка реализованы при помощи паттерна “Снимок”
- Реализован контроль корректности файла с сохраненными данными

## **Выполнение программы**

Загрузка и сохранения реализованы по принципу паттерна «Снимок». Класс Game возвращает снимок(GameMemento) при считывании состояния программы в методе readMemento(string), а так же создает снимок при записи состояния в методе restoreMemento(Memento \*). Для передачи информации использованы дополнительные классы.

Для передачи и сохранения различных состояний игры введены дополнительные классы: GameParam(содержащий значение ширины, высоты, массив информации о базах и нейтральных объектов, массив ландшафта поля), BaseParam(содержащий номер базы, количество юнитов базы, максимальное количество юнитов, здоровье и координаты базы), UnitParam(содержащий название юнита, базу к которой он принадлежит и значение атрибутов юнита), NeutralParam(тип и координаты).

Для сохранения и загрузки в файл по идиоме RAII созданы классы WriteToFile, ReadFromFile, которые проверяют корректность файла с сохраненными данными.

- В классе WriteToFile для чтения из файла используется класс ifstream, в ReadFromFile для записи – std::ofstream.
- В обоих классах файл открывается в конструкторе (при ошибке бросается соответствующее исключение), а закрывается в деструкторе.

Контроль корректности файла реализован следующим образом:

- Проверка на наличие заголовка и контроль корректности значений осуществляется в методе read() класса MementoReader.

Класс GameMemento реализован по принципу *паттерна «Снимок»*.

Паттерн «Снимок» также предполагает наличие «Создателя», которым является класс Game в котором были добавлены методы для сохранения и загрузки игры.

## **Выводы**

В ходе выполнения лабораторной работы была написана программа, в которой реализованы классы для сохранения и загрузки состояния программы. Были использованы паттерны проектирования, а также принципы объектно-ориентированного программирования.