**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: Сериализация, исключения

|  |  |  |
| --- | --- | --- |
| Студент гр. 1384 |  | Алиев Д.А. |
| Преподаватель |  | Жангиров Т.Р. |

Санкт-Петербург

2022

**Цель работы.**

Разработать классы, позволяющие задать клавиши для управления. Разработать классы, обрабатывающие команды пользователя таким образом, чтобы без изменения существующего кода можно было внедрить иной способ ввода команд.

**Задание.**

Реализовать систему классов позволяющих проводить сохранение и загрузку состояния игры. При загрузке должна соблюдаться транзакционность, то есть при неудачной загрузки, состояние игры не должно меняться. Покрыть программу обработкой исключительных состояний.

**Требования:**

* Реализована загрузка и сохранение состояния игры
* Сохранение и загрузка могут воспроизведены в любой момент работы программы.
* Загрузка может произведена после закрытия и открытия программы.
* Программа покрыта пользовательскими исключениями.
* Пользовательские исключения должны хранить полезную информацию, например значения переменных при которых произошло исключение, а не просто сообщение об ошибке. Соответственно, сообщение об ошибке должно учитывать это поля, и выводить информацию с учетом значений полей.
* Исключения при загрузке обеспечивают транзакционность.
* Присутствует проверка на корректность файла сохранения. (Файл отсутствует; в файле некорректные данные, которые нарушают логику; файл был изменен, но данные корректны с точки зрения логики).

**Примечания:**

* Исключения должны обрабатываться минимум на фрейм выше, где они были возбуждены
* Для реализации сохранения и загрузки можно использовать мементо и посетителя
* Для проверки файлов можно рассчитывать хэш от данных.

**Выполнение работы.**

Было разработано 3 интерфейса и их реализации:

1. SaveLoader – интерфейс, отвечающий за загрузку сохранения
   1. SaveFileLoader –класс, отвечающий за загрузку сохранения из файла
2. MapSaver – интерфейс, отвечающий за сохранение карты
   1. FileMapSaver – класс, отвечающий за сохранение карты в файл
3. SaveableComponent – интерфейс, отвечающий за то, что объекты являются сериализуемыми. Данный интерфейс также является частью паттерна проектирования «Компоновщик».

Для реализации исключений были разработаны классы-наследники интерфейса exception стандартной библиотеки: SlonException и LoadingMapException. Также был разработан класс StringUtilities, имеющий набор функций, необходимых для обработки строк.

За сохранение в файл ответственен класс FileMapSaver – он просто записывает строку в необходимый файл. Строка берётся из метода toString объекта поля. Метод возвращает строку, содержащую информацию о размерах карты и о всех клетках поля – для этого у клеток вызывается такой же метод toString, который, в свою очередь ещё вызывает метод toString события, находящегося в клетке. Также в строку записывается информация о положении игрока и после о его характеристиках – опять же используя метод toString игрока. Любой объект представляется как имя объекта и набор параметров в виде param1={data1}. Таким образом, по такому представлению можно будет восстановить объект.

Загружает сохранение из файла класс SaveFileLoader. Считывая строки, он вызывает статические методы fromString создаваемых объектов (клетки, игрока) и записывает их в объект поля.

Диаграмма разработанных в ходе лабораторной работы классов:

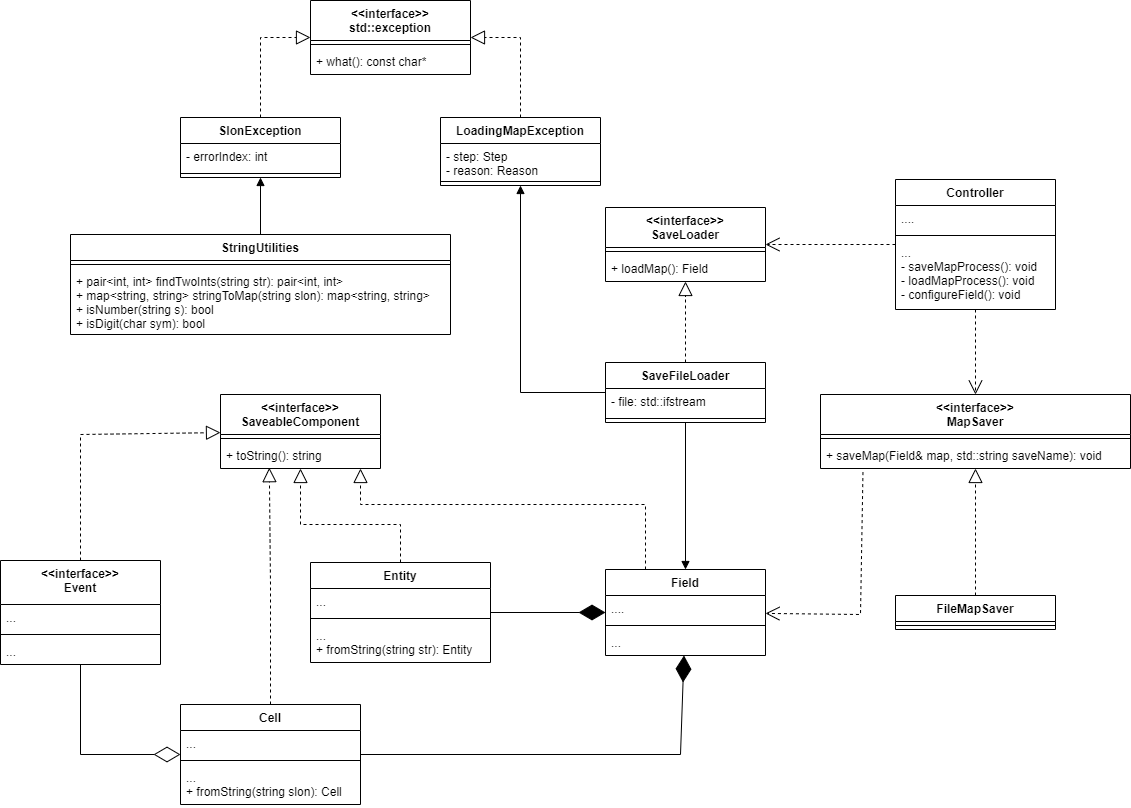


Рисунок 1 – Диаграмма классов

**Тестирование.**

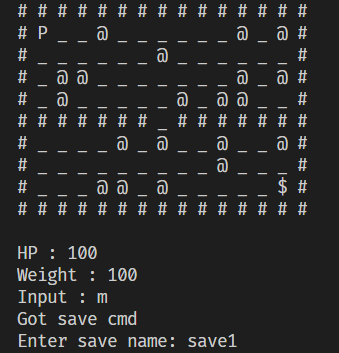


Рисунок 2 – Cохранение начальной позиции

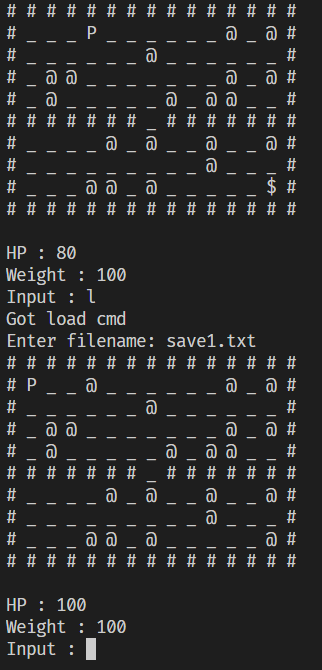


Рисунок - прошлись и получили урон, откатились

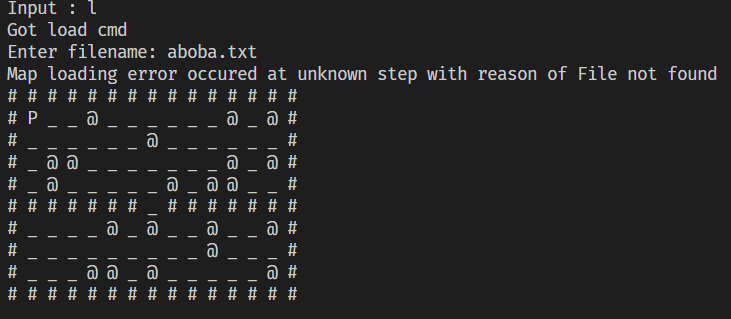


Рисунок - попытка загрузки из несуществующего файла

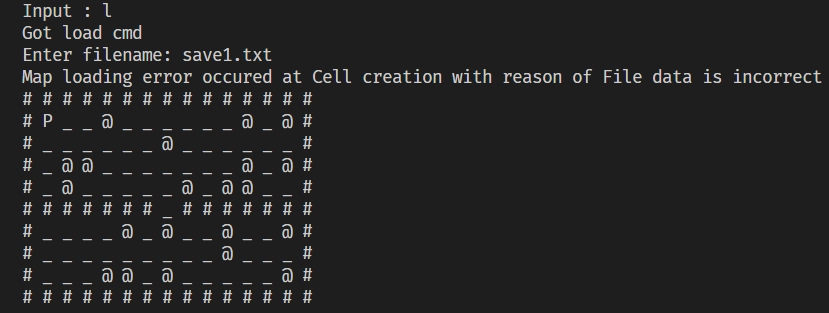


Рисунок - попытка загрузки повреждённого файла

**Выводы.**

В ходе выполнения лабораторной работы были разработаны классы, позволяющие сохранить и загрузить прогресс выполнения программы. Также программа была покрыта обработкой исключительных ситуаций.