**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №3**

**по дисциплине «Объектно-Ориентированное программирование»**

Тема: Логирование, перегрузка операций

|  |  |  |
| --- | --- | --- |
| Студентка гр. 1303 |  | Королева П.А |
| Преподаватель |  | Жангиров Т.Р |

Санкт-Петербург

2022

## Цель работы.

## Реализовать класс/набор классов отслеживающих изменения состояний в программе.

## Задание.

Реализовать класс/набор классов отслеживающих изменения состояний в программе. Отслеживание должно быть 3-х уровней:

* Изменения состояния игрока и поля, а также срабатывание событий
* Состояние игры (игра начата, завершена, сохранена, и.т.д.)
* Отслеживание критических состояний и ошибок (поле инициализировано с отрицательными размерами, игрок попытался перейти на непроходимую клетку, и.т.д.)

Реализованы классы для вывода информации разных уровней для в консоль и в файл с перегруженным оператором вывода в поток.

## Выполнение работы.

Для реализации системы логирования были разработаны классы:

Message – интерфейс сообщения, которое содержит в полях некую информацию о состоянии программы. Его реализуют GameMassage, ModelMessage, ErrorMessage. Каждый тип сообщения имеет разное количество полей, тк разные уровни логирования требуют разную степень детализации информации. Для каждого класса перегружен оператор вывода в поток.

Log – интерфейс, используемый для создания сообщений Message. Его реализуют классы GameLog, ModelLog и ErrorLog, каждый из которых создает Message своего типа.

Logger – интерфейс для вывода сообщений в консоль или в файл. Его реализуют классы FileLogger и ConsoleLogger. Они получают указатель на Message, с помощью dynamic\_cast приводят к нужному типу и выводят в соответствующий поток.

LoggerPool – главный связующий класс в системе логирования. В него передаются логи из основной программы, создаются сообщения в соответствии с уровнем логирования, выбранным пользователем, и выводятся в поток.

Класс реализует паттерн синглтон. Это позволяет любым классам из любой части программы обращаться к системе логирования и передавать свое текущее состояние.

LoggerPoolManager ― в нем происходит взаимодействие с пользователем: вывод на экран доступных уровней и типов логирования и считывание выбора пользователя, происходит инициализация LoggerPool.

**UML-диаграмма иерархии событий.**

Иерархия классов событий представлена на рисунке 1.

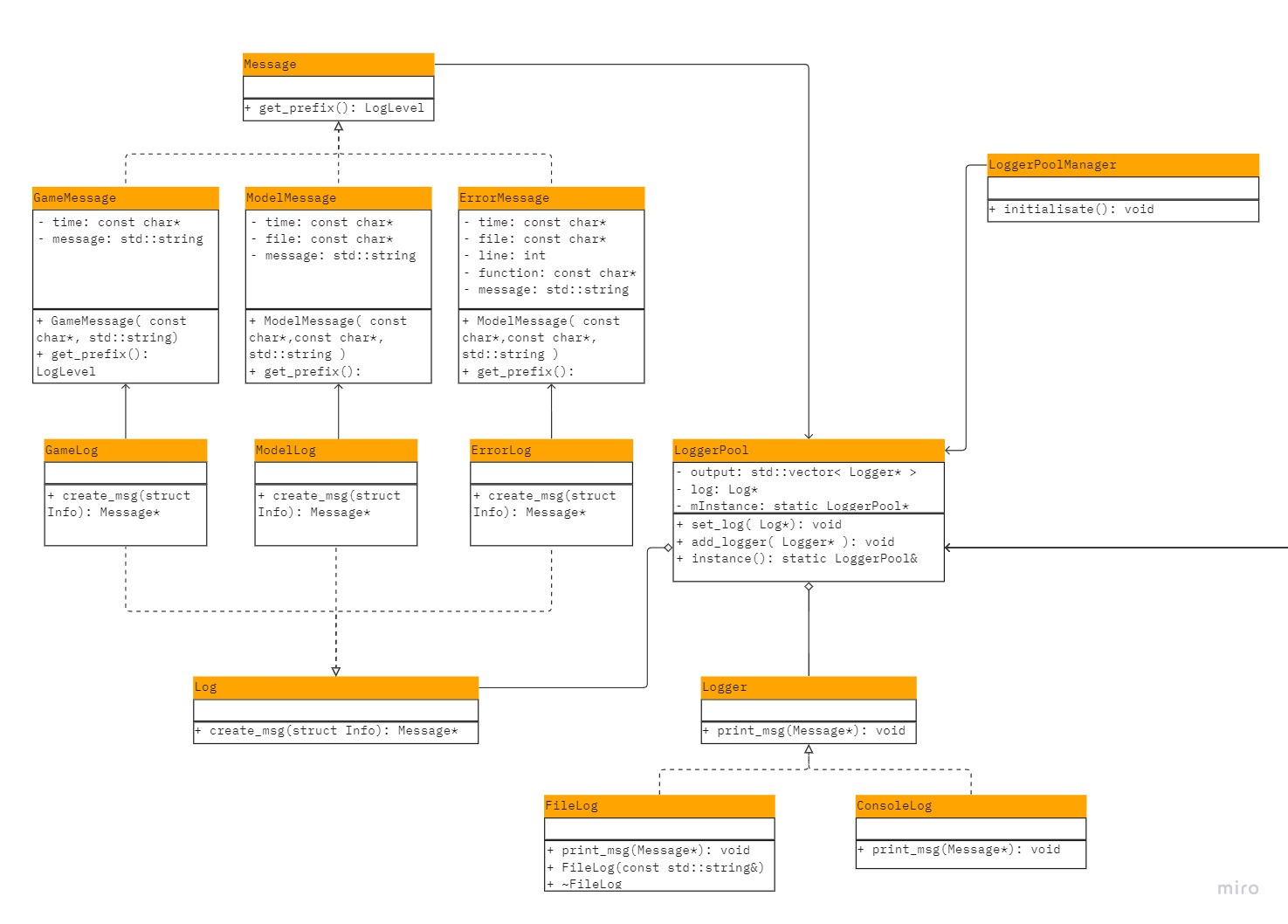


Рис.1.1 – Диаграмма иерархии классов логирования

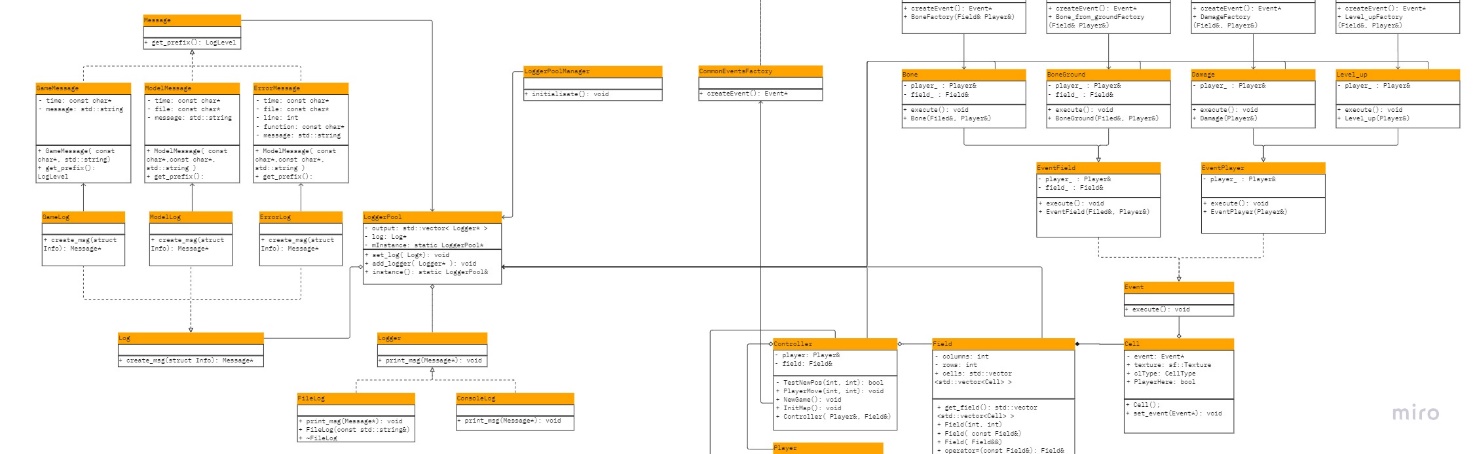
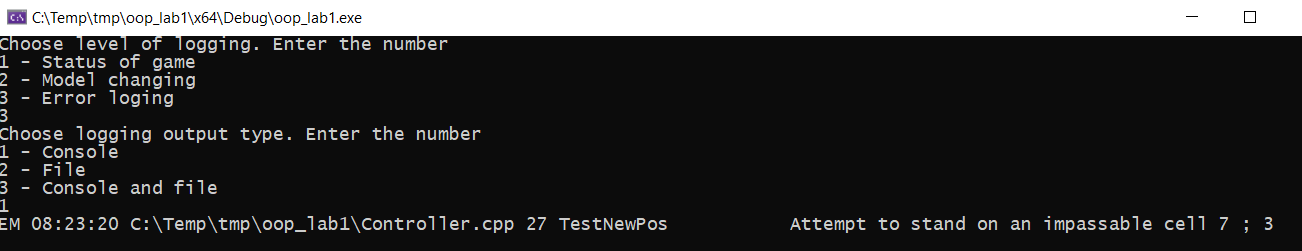


Рис.1.2 – Связь системы логирования и основной программы

## Тестирование.



## Рис.2 – Запуск программы, задание уровня и типа логирования

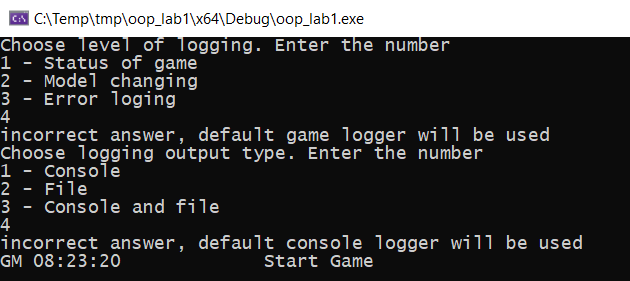


Рис.3 - Обработка неправильных введенных данных – значения по умолчанию

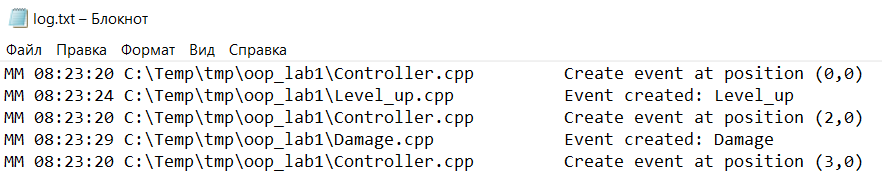


Рис.4 – Логирование в файл

## Выводы.

## Был реализован класс/набор классов отслеживающих изменения состояний в программе.