

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: Перегрузка операторов / Логирование

Студент гр. 2300

Жохов К.С.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2023

Цель работы.

Разработать систему классов, которая будет отслеживать происходящие события в игре и логировать их.

Задание.

а) Реализовать набор классов “сообщений” с общим интерфейсом, который будут срабатывать в определенные моменты и хранить информацию о событии, но не должны хранить сообщение в виде строки. Должны быть реализованы класс для следующих событий:

- Игрок выиграл. Хранится информация о характеристиках игрока
- Игрок проиграл. Хранится информация о координатах клетки на которой событие произошло
- Была запущена новая игра. Хранится информация о размерах поля и стартовой позиции игрока.
- Была введена клавиша и сработала команда. Информация о введенном символе и какая команда сработала.
- Была введена клавиша, но никакая команда не сработала. Информация о введенном символе.

б) Для сообщений перегрузить оператор вывода в поток. Таким образом можно выводить сообщение в различные потоки (cout, файл). При выводе в поток сообщения, должна формироваться строка и подставляться хранимая информация.

в) Разработать систему классов, которые отслеживают сообщения и выводят их в файл и/или консоль. Куда выводить запрашивается у пользователя при запуске программы: никуда, в файл, в консоль, в файл и консоль. Классы, в которых происходит отслеживаемое событие, должны только отправлять сообщение, но не знать куда, то есть только создают сообщение, инициализируя его информацию, и отправляют.

Выполнение работы.

1. Класс `Message`. Представляет собой интерфейс сообщения. Содержит чисто виртуальный `public` метод, который возвращает сообщение в виде строки.

2. От класса `Message` унаследованы классы `MessageWin`, `MessageLose`, `MessageCommandKey`, `MessageUnknownKey`, `MessageNewGame`, которые представляют собой реализацию интерфейса сообщения. Каждый из них содержит информацию о соответствующем событии в игре.

3. Класс `Logger`. Представляет собой интерфейс логера. Содержит виртуальный метод, который выводит, полученное логером сообщение в поток вывода.

4. От класса `Logger` унаследованы классы `ConsoleLogger` и `FileLogger`. Представляет собой реализацию интерфейса логера, способны выводить сообщения в консоль и файл соответственно.

5. Теперь `GameManager` содержит в полях вектор логеров, которые добавляются в игру после решения пользователя о применении логирования в игре. Также добавлен метод `sendLog()`, который принимает на вход объекта класса сообщения и отправляет его всем логерам игры с целью получения от них логов.

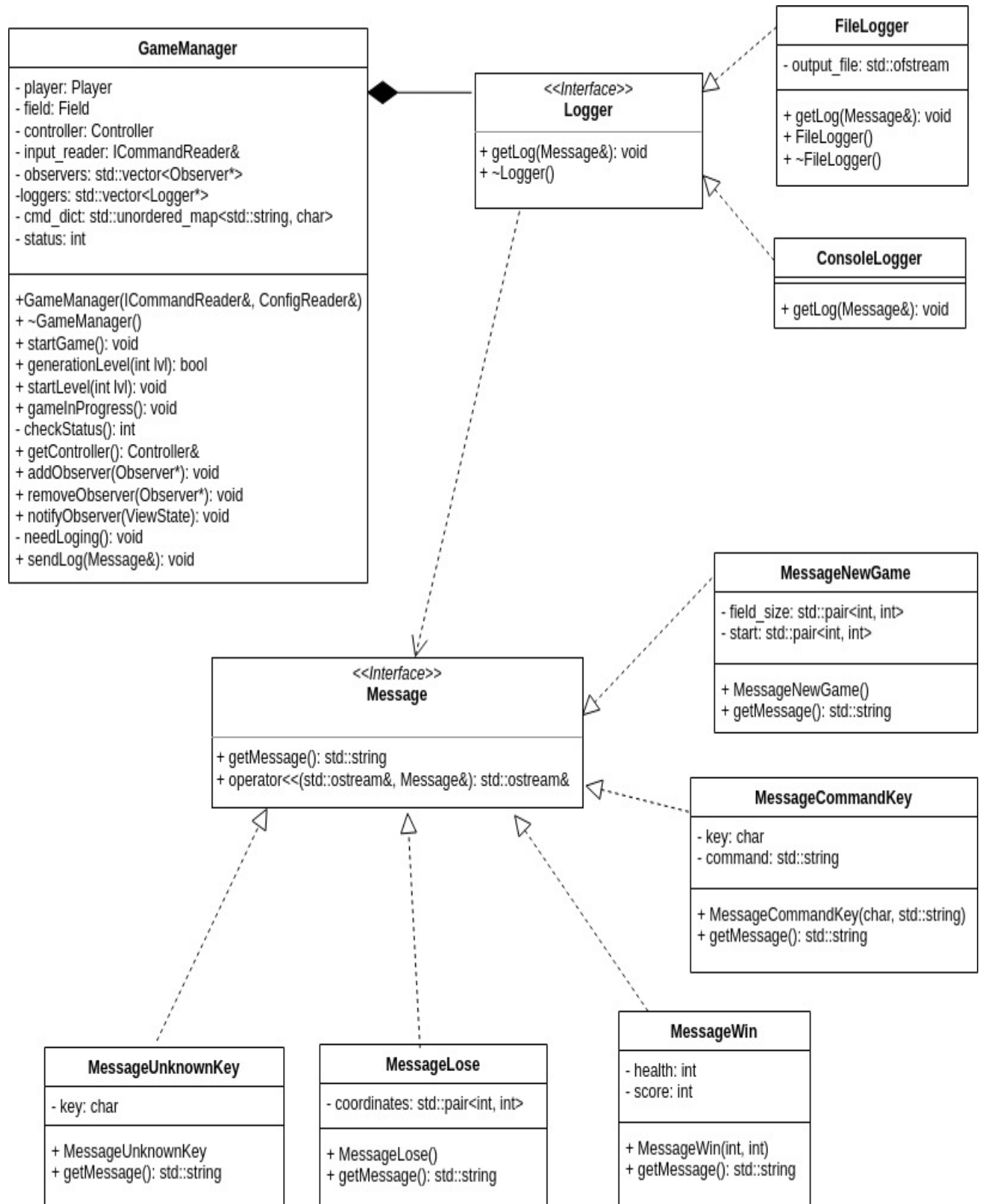
Разработанные UML-диаграммы классов см. в приложении А.

Результаты тестирования см. в приложении Б.

Выводы.

В ходе лабораторной работы была разработана система логирования игры. По желанию пользователя логи могут выводиться в консоль и файл.

ПРИЛОЖЕНИЕ А **UML-ДИАГРАММЫ КЛАССОВ**



ПРИЛОЖЕНИЕ Б

ТЕСТИРОВАНИЕ

Таблица Б.1 - Примеры тестовых случаев

№ п/п	Входные данные	Выходные данные	Комментарии
1.	2 1 wwdq q	A new game has been launched Field size: 5, 5Player's starting position: 0, 0 The key 'w' was entered and the 'UP' command worked The key 'w' was entered and the 'UP' command worked The key 'd' was entered and the 'RIGHT' command worked The key 'q' was entered and the 'QUIT' command worked	Верный результат
2.	1 2 ssaaaa sssss q	A new game has been launched Field size: 10, 10 Player's starting position: 0, 9 The key 's' was entered and the 'DOWN' command worked The key 's' was entered and the 'DOWN' command worked The key 'a' was entered and the 'LEFT' command worked The key 'a' was entered and the 'LEFT' command worked The key 'a' was entered and the 'LEFT' command worked The key 'a' was entered and the 'LEFT' command worked The key 'a' was entered and the 'LEFT' command worked The key 's' was entered and the 'DOWN' command worked The key 's' was entered and the 'DOWN' command worked The key 's' was entered and the 'DOWN' command worked The key 's' was entered and the 'DOWN' command worked The key 's' was entered and the 'DOWN' command worked DEFEAT! Coordinates: 0, 4	Верный результат
3.	3 1 wwdaw wdddd q	A new game has been launched Field size: 5, 5Player's starting position: 0, 0 The key 'w' was entered and the 'UP' command worked The key 'w' was entered and the 'UP' command worked The key 'd' was entered and the 'RIGHT' command worked	Верный результат

		<p>The key 'a' was entered and the 'LEFT' command worked</p> <p>The key 'w' was entered and the 'UP' command worked</p> <p>The key 'w' was entered and the 'UP' command worked</p> <p>The key 'd' was entered and the 'RIGHT' command worked</p> <p>The key 'd' was entered and the 'RIGHT' command worked</p> <p>The key 'd' was entered and the 'RIGHT' command worked</p> <p>The key 'd' was entered and the 'RIGHT' command worked</p> <p>VICTORY! Health: 100 Score: 0</p>	
4.	<p>2</p> <p>2</p> <p>ftddss</p> <p>q</p> <p>q</p>	<p>A new game has been launched Field size: 10, 10 Player's starting position: 0, 9</p> <p>An unknown key 'f' was entered - no command worked</p> <p>An unknown key 't' was entered - no command worked</p> <p>The key 'd' was entered and the 'RIGHT' command worked</p> <p>The key 'd' was entered and the 'RIGHT' command worked</p> <p>The key 'd' was entered and the 'RIGHT' command worked</p> <p>The key 's' was entered and the 'DOWN' command worked</p> <p>The key 's' was entered and the 'DOWN' command worked</p> <p>The key 'q' was entered and the 'QUIT' command worked</p>	Верный результат