

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.

Студент гр. 0381

Дзаппала Д.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2021

Цель работы.

Понять что такое логирование и лог-файлы. Реализовать класс логгера, который будет записывать в консоль, файл или в консоль и в файл. Должна соблюдаться идиома RAII.

Задание.

Необходимо проводить логирование того, что происходит во время игры.

Требования:

- Реализован класс логгера, который будет получать объект, который необходимо отслеживать, и при изменении его состояния записывать данную информацию.
- Должна быть возможность записывания логов в файл, в консоль или одновременно в файл и консоль.
- Должна быть возможность выбрать типа вывода логов
- Все объекты должны логироваться через перегруженный оператор вывода в поток.
- Должна соблюдаться идиома RAII

Потенциальные паттерны проектирования, которые можно использовать:

- *Адаптер (Adapter) - преобразование данных к нужному формату логирования*
- *Декоратор (Decorator) - форматирование mBBЦекста для логирования*
- *Мост (Bridge) - переключение между логированием в файл/консоль*
- *Наблюдатель (Observer) - отслеживание объектов, которые необходимо логировать*
- *Синглтон (Singleton) - гарантия логирования в одно место через одну сущность*
- *Заместитель (Proxy) - подстановка и выбор необходимого логирования*

Выполнение работы.

Был создан шаблонный класс `Logger`, шаблоном которого является переменная перечисления `LoggerType`. `LoggerType` обладает перечислениями `FILE`, `CONSOLE` и `BOTH`, чтобы можно было записывать логи в файл, консоль или оба местоположения, соответственно. Класс содержит одно поле — объект `std::ofstream outFile`, поток вывода в файл. В конструкторе инициализируется поле `outFile`, запись логов идет в файл `log.txt`. В классе определены три метода. Метод `log` перегружен. Один из них является шаблонным, принимающий ссылку на объект. Второй обычный, принимающий ссылку константный объект `std::string`, нужен в том случае, если объект логгера будет закинут в какой-нибудь метод, и оттуда напрямую залогировать что-нибудь. В `log` выводятся координаты объекта врага или игрока, а также их текущий уровень здоровья. Также существует метод `logTime`, который вызывается перед тем, как логировать объекты, так как этот логгер выводит дату и время лога.

Так как появился логгер, в глобальном классе `Game` теперь есть указатель на логгер, а также метод `PrintLog()`, который печатает лог для Игрока и всех врагов.

Выводы.

Был реализован класс логгера, который может записывать данные в файл, консоль, или сразу в оба места. Идиома RAII соблюдается.

ПРИЛОЖЕНИЕ А ИСХОДНЫЙ КОД ПРОГРАММЫ

