

**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: Логгирование, перегрузка операторов.**

Студент гр. 0303

\_\_\_\_\_

Болкунов В.О.

Преподаватель

\_\_\_\_\_

Жангиров Т.Р.

Санкт-Петербург 2021

### **Цель работы.**

Создать набор классов для осуществление логгирования объектов игры.

### **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 состоянии 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

### **Основные теоретические положения.**

Логгирование - форма автоматической записи в хронологическом порядке операций в информационных технологиях, процесс записи информации о происходящих в рамках какого-либо процесса с некоторым объектом событиях, например, в файл регистрации или в базу данных.

### **Выполнение работы.**

Структура **Level** — хранит в себе перечисление уровней логов, по совместительству объект структуры определяет уровень.

Структура **Log** — объект лога, хранит сообщение, временной штамп и уровень лога. Для удобного создания лога каждого уровня определены макросы:

- **error(msg)**
- **warn(msg)**
- **info(msg)**
- **debug(msg)**

Класс **LogTimer** — синглтон для отсчёта начального времени логов (точка отсчёта).

Класс **Format** — абстрактный декоратор для логов. Содержит виртуальный метод `wrap` для обёртки лога.

Класс **EmptyFormat** — формат по умолчанию (не меняет лог)

Класс **TagFormat** — формат с тэгом в квадратных скобках.

Класс **TimeFormat** — формат с временной меткой в квадратных скобках, время = разность между меткой лога и точкой отсчёт **LogTimer**-а.

Класс **Logger** — абстрактный класс для логгеров, принимает логи через перегруженный оператор `<<`, имеет свой объект форматирования.

Класс **FileLogger** — логгер вывода в файл, принимает имя файла. Время жизни файла контролируется объектом логгера.

Класс **ConsoleLogger** — логгер вывода в консоль, принимает ссылку на поток вывода.

Класс **Loggable** — абстрактный класс логируемого объекта, содержит множество указателей на логгеры, методом **notify** выводит лог во все имеющиеся у него логгеры. Реализует паттерн Observer.

Разработанный программный код см. в директории */game\_lib*.

Разработанную диаграмму классов UML см. в Lab1\_UML.pdf .

### **Тестирование.**

Разработанные тесты см. в директории */tests*.

### **Выводы.**

Был реализован набор классов для осуществления удобного логгирования и в игровые объекты добавлены выводы логов.