



МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ М. В. ЛОМОНОСОВА
Факультет вычислительной математики и кибернетики
Кафедра алгоритмических языков

Отчёт о выполнении задания практикума

«Разработка игры Platformer»

Студент 324 группы
И. А. Буримский

Москва, 2019

1 Постановка задачи, основной функционал приложения

1.1 Описание

Платформер — жанр компьютерных игр, в которых основной чертой игрового процесса является прыгание по платформам, лазанье по лестницам, собирание предметов, обычно необходимых для завершения уровня.

В процессе игры у главного героя заканчивается жизнь. Ему необходим удар электрическим током, чтобы пополнить ХР. Для этого используются батарейки и трансформаторы. Батарейки встречаются часто, но они дают небольшой прирост. Трансформаторы встречаются редко, но прирост жизни от них значительнее. Также главного героя преследует враг, который убивает его, когда догоняет. В процессе игры подсчитывается пройденная дистанция.

1.2 Базовая часть

Приложение:

1. игровой мир генерируется автоматически;
2. есть возможность проиграть или двигаться дальше;
3. предоставлено минимальное управление персонажем;
4. определяется момент поражения и подсчитывается пройденная дистанция.

2 Структура проекта

2.1 Модули, их взаимосвязь

Исходный код находится в следующий файлах:

- **Main.hs** - программа, запускающая игру.
- **Platformer.hs** - модуль, запускающий функцию play.
- **CommonFunctions.hs** - модуль, хранящий вспомогательные функции, необходимые в нескольких модулях.
- **Constants.hs** - модуль, хранящий константы.
- **DrawFunctions.hs** - модуль, рисующий карту.
- **HandleFunctions.hs** - модуль, отлавливающий нажатия клавиш.
- **InitFunctions.hs** - модуль, инициализирующий игру.
- **Types.hs** - модуль, объявляющий типы.
- **UpdateFunctions.hs** - модуль, обновляющий состояние игры.

2.2 Проект

- В папке app - находится Main.hs. Папка src - содержит все необходимые модули.
- Папка img - содержит все картинки, использованные в игре.

3 Используемые библиотеки

- *Graphics.Gloss* - для отображения графического интерфейса.
- *System.Random* - для получения случайного направления полёта шарика и цвета платформ.
- *System.IO* - для работы с файлами.
- *Graphics.Gloss.Juicy* - для добавления изображений.

4 Сценарии работы с приложением и примеры использования

Проект можно собрать и запустить при помощи команды **stack build && stack exec project-template-exe**.

При запуске приложения игрок начинает убегать от врага. Ему необходимо подпрыгивать, чтобы забраться на кубики выше. Для того, чтобы подпрыгнуть, нужно нажать на пробел.