|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА - Российский технологический университет»**  **РТУ МИРЭА** |

РТУ МИРЭА

Кафедра Интеллектуальные системы информационной безопасности (КБ-4)

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К ПРОЕКТУ**

**«ПИНГ-ПОНГ»**

по дисциплине

«Интерпретируемый язык программирования высокого уровня»

Выполнила студентка группы БСБО-07-19 Саблина Е.В.

Принял

Преподаватель Тарланов А.Т.

Москва 2021

1) Создать свою реализацию классической аркадной игры Пинг-Понг на интерпретируемом языке программирования Python с использованием библиотеки tkinter.

2) Чтобы создать окно, в котором будет отображаться графика, используем классы библиотеки Tkinter. Чтобы создать окно, где будет видна графика, был использован класс Tk(). Он просто делает окно, но без содержимого. Чтобы появилось содержимое, необходимо создать холст — видимую часть окна. С помощью класса Canvas() был создан объект холста.

В начале игры мячик располагается посередине поля. Затем он летит в сторону случайного игрока. Задача каждого игрока – отбить шарик платформой. Каждый раз, когда один из игроков отбивает шарик, скорость его движения возрастает. Если игрок пропустил мяч за свою линию, счет соперника увеличивается, а скорость движения мяча сбрасывается.

3) Для графики в игре используем библиотеку Tkinter. Она входит в набор стандартных библиотек Python и позволяет рисовать простейшие объекты — линии, прямоугольники, круги и красить их в разные цвета.

Управление осуществляется с помощью клавиатуры. Первый пользователь – клавиши «W», «S», второй – стрелки «вверх» и «вниз». Обработка их нажатия и отпускания производится с помощью функций movement\_handler(event) и stop\_pad(event) соответственно. Далее эти функции привязываются методом bind() библиотеки Tkinter к нашему объекту Canvas. Движение мяча реализовано в функции move\_ball(), которая вызывает рекурсивно саму себя через root.after, движение платформ - в move\_pads(), обработка отскока мяча от платформ - в bounce(). Скорость движения мяча по осям X и Y – на сколько пикселей сместится шарик относительно своего текущего положения.

Для направления и скорости мяча по оси Y используем модуль random, входящий в состав стандартной библиотеки языка Python. Он содержит множество функций, связанных с эмуляцией случайности. В данном проекте модуль random был использован для того, чтобы случайным образом задавать направление движения мяча (например, по оси Y при отбитии ракеткой). Также с помощью метода random.random() слегка меняем исходное направление мяча после забитого гола, чтобы было интересней играть.

