**Постановка задачи**

Задача: ознакомиться с Unity и научиться делить 3D игры в нём.

Суть программы: В главном меню программы пользователь выбирает игру и играет в неё. Сами игры очень понятны (наподобие «Сапёра», «Змейки», «Арканоида» и т.д.) и приятны для глаза.

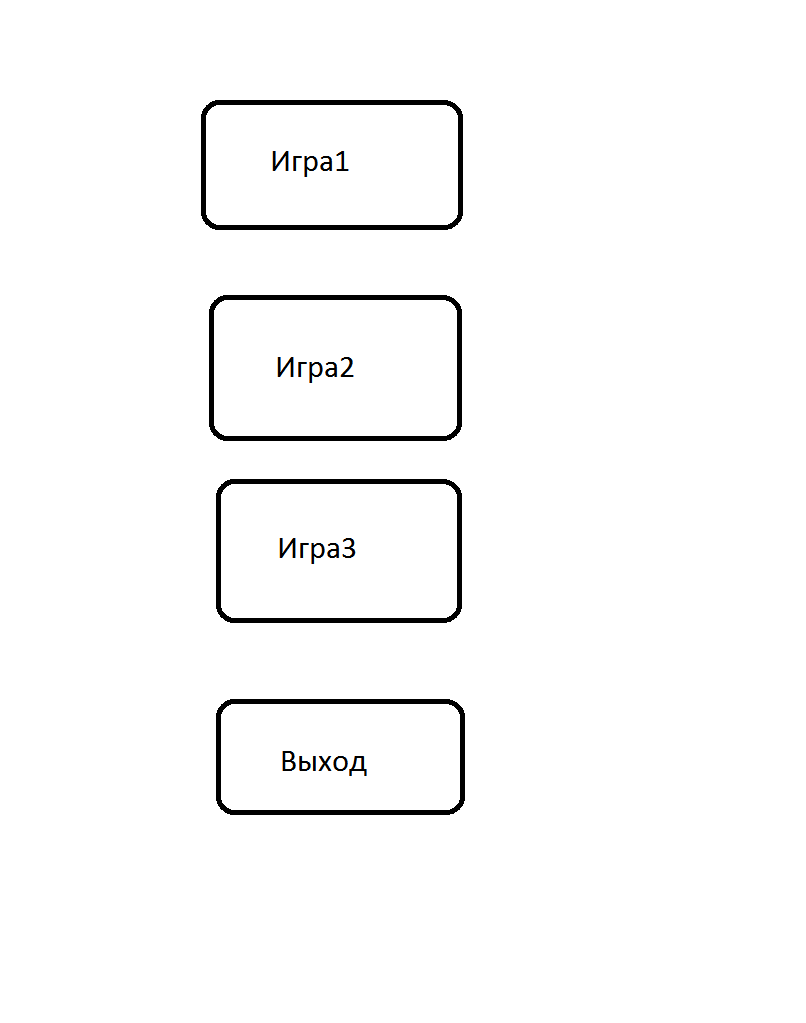
Сложность проекта: Средняя. Лёгкость игр компенсируется их количеством.

Кто может быть заинтересован в программе: Офисные сотрудники. Пользователи слабых по мощности компьютеров.

Перспективы: Со временем можно добавлять новые игры.

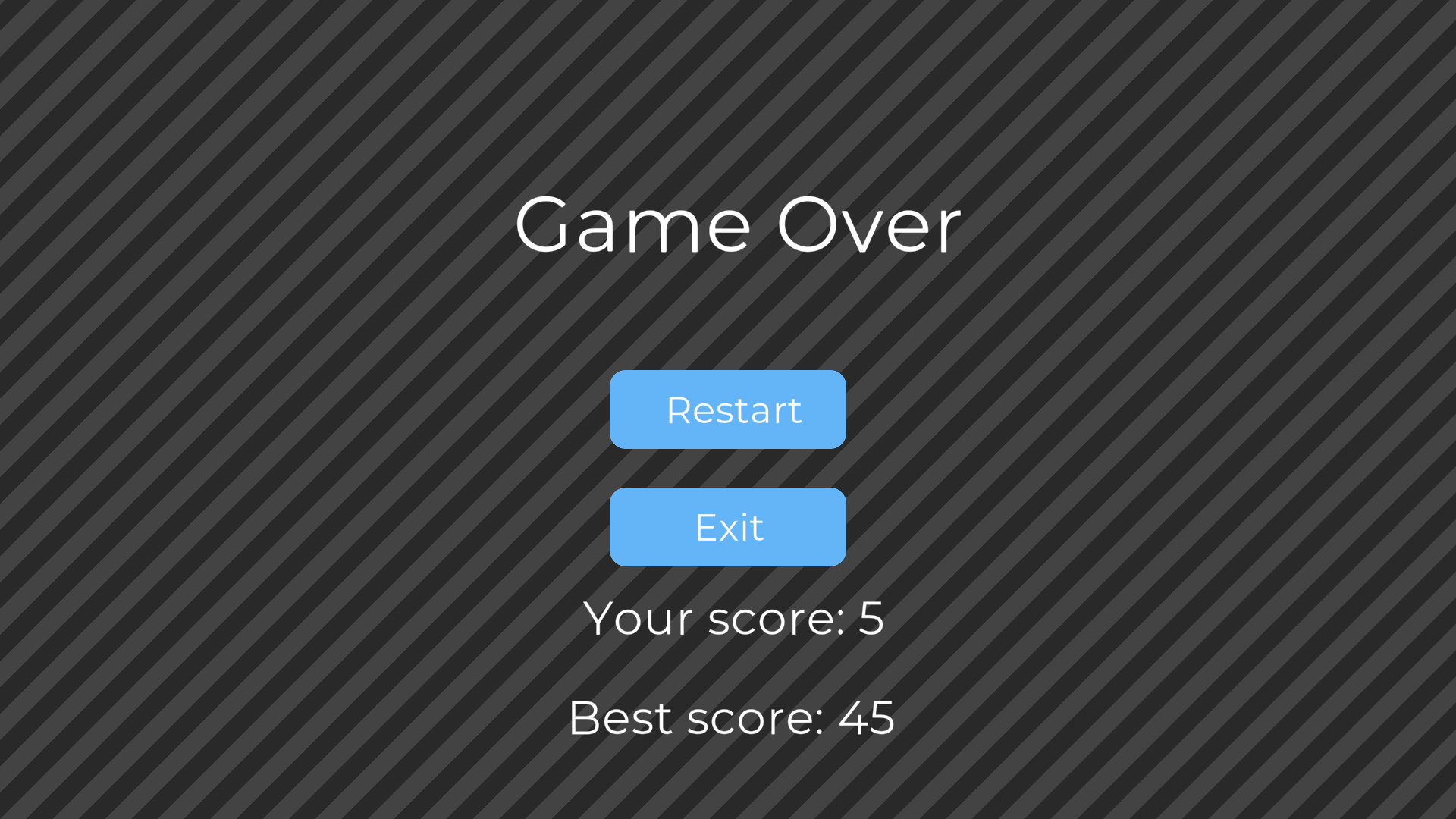
Среда разработки: Unity, Visual Studio.

Язык программирования: C#.



**Входные и выходные данные**

Программа будет выводить количество очков за игру и рекордное количество очков, хранимое в памяти компьютера в виде целочисленной переменной.



**Математическая модель**

* Flappy bird

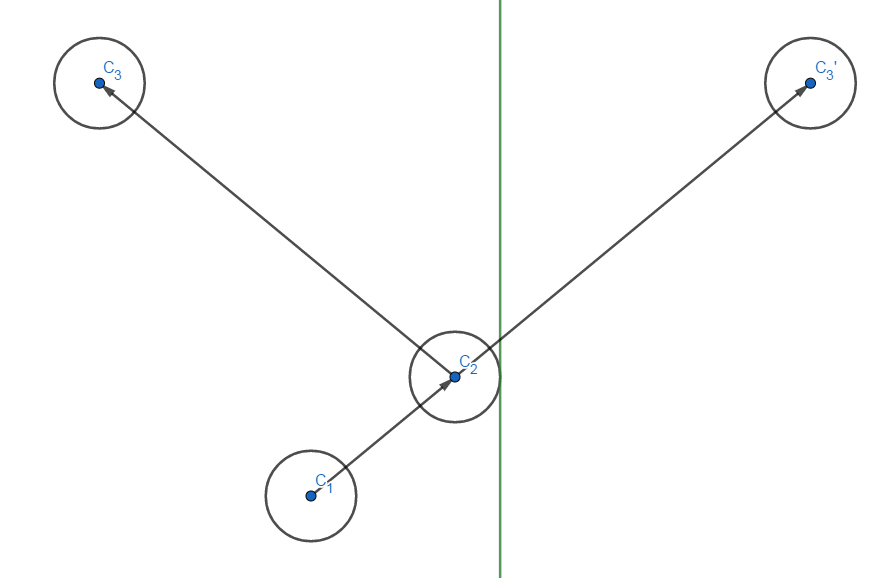
Ускорение, придаваемое птице по нажатию на пробел вычисляется по следующей формуле:

Force = force – velocity

Где velocity – вертикальная скорость птицы, force – ускорение при нулевой вертикальной скорости, Force – конечное ускорение.

* Arkanoid

Угол падения мяча равен углу отражения



**Выбор структуры данных**

* Snake

В динамическом массиве Tails типа Transform хранится информация о ячейках, из которых состоит тело змеи.

* Sapper

Каждая клетка имеет динамический массив adjacentTiles типа Tile, в котором хранится информация о соседних клетках. Также каждая клетка имеет свой собственный ID номер в виде целочисленной переменной и логическую переменную с информацией о мине.

Поле, в котором создаются все клетки, имеет массив tilesAll типа Tiles, содержащий все клетки и два динамических массива tilesUnmined и tilesMined типа Tile, хранящие информацию о заминированных и разминированных клетках.

* Arkanoid

Каждый блок хранит в виде целочисленной переменной количество оставшихся ударов.

**Выбор метода решения**

* Flappy bird

Птица движется только вдоль одной оси, а окружающие объекты – по двум осям, таким образом появляется ощущение, что птица летит. За пределами экрана появляются облака и препятствия, затем они проходят через экран, удаляются и создаются заново.

* Arkanoid

Перед началом игры блоки генерируют свои типы. Чем больше ударов нужно для разрушения, тем меньше вероятность, что выберется именно такой тип блока. С помощью Unity настраиваем физику мяча.

* Sapper

Перед началом игры клеткам присваиваются свои ID номера. Затем с помощью этих номеров каждая клетка определяет состояние соседних клеток.

**Построение алгоритма**

Snake

****

Sapper 