Тренировочное задание

Для подготовки к городскому конкурсу

«Шаг в профессию»

Компетенция «Программирование игр»

Младшая возрастная категория 10+ (10 -13 лет)

**Описание возможного варианта конкурсного задания**

Работа выполняется на языке программирования Scratch, разработанной в Массачусетском технологическом институте. Одной из отличительных особенностей объектно-ориентированного языка программирования Scratch является наличие библиотеки персонажей и фонов игрового поля, которые могут быть запрограммированы на выполнение операций и действий.

 Задание исключает этап подготовки, в который входит задача разработки концепции игры, дизайна персонажей, выбора средств реализации проекта, подготовки плана, по которому будет создаваться игра.

 Одной из задач проведения соревнований является погружение участников в процесс производства компьютерных игр, в ходе которого необходимо проявить умения и навыки, востребованные во множестве «игровых» профессий, таких как: «Программист», «Художник», «Музыкант», «Писатель», «Дизайнер», «Тестер» и т.д.

Участникам соревнований предлагается самостоятельно выполнить проект компьютерной игры, состоящей из 3-х мини-игр. Проект представляет интеграцию мини-проектов, где есть начальный экран, через объекты которого игрок может перемещаться в область одной из следующих мини-игр:

 «Snake»;

«Rocket»;

«Arcanoid»;

 **Содержание задания**

**Snake**

Описание локации:

 В центре игрового поля располагается спрайт Snake, в виде головы змеи. Управление Snake осуществляется в следующих направлениях: вверх, вниз, направо, влево. Туловище образуется при помощи инструмента «Перо». За спрайтом Snake с определённой задержкой следует Спрайт-ластик, костюм которого совпадает с цветом фона. В координатах с произвольными значениями на экране появляется Спрайт-добыча. При касании Snake Спрайта-добычи, происходит увеличение задержки движения Спрайта-ластика, тем самым увеличивается длина туловища Snake. После касания Спрайт-добыча прячется и появляется в новом месте на экране. В результате 10 касаний Спрайта-добычи игра переходит в режим «Пройденной». При касании Snake края экрана игра начинается сначала.

Примечание: Для организации синхронного движения Snake и Спрайта-ластика необходимо создать два массива данных (Списки) для координат Х и Y. Перемещаясь по экрану, Snake заполняет массивы, отправляя данные о своём положении по оси Х и Y. Спрайт-ластик считывает данные о координатах и занимает положение на экране в соответствии этими координатами. Для перехода к считыванию следующей пары данных о координатах Snake необходимо создать переменную, которая будет циклично прирастать на единицу сразу же после считывания предыдущих данных о координатах.

Программирование объектов:

*Snake*

* По запуску локации переходит в центр экрана координаты по оси Х – 0, по оси Y – 0;
* Управление с помощью →←↑↓

или WASD

ВАРИАНТ СКРИПТА

* Использует перо определённого цвета и размера для рисования туловища;

ВАРИАНТ СКРИПТА

* Выполняется условие: при касании края экрана – запуск игры сначала;

ВАРИАНТ СКРИПТА

* Выполняется условие: при касании элемента головы (красный цвет) своего туловища (синий цвет) – запуск игры сначала;

ВАРИАНТ СКРИПТА

Примечание: для осуществления выполнения этого условия рекомендуется при рисовании костюма Snake включить элемент – язык змеи, цвет которого отличен от цвета головы змеи. В этом случае возможно организовать выполнение условия через блок Сенсора

* Очистка экрана от результатов рисования пером от предыдущих запусков локации;
* Удаление данных из списков с координатами по оси Х и Y;
* Цикл заполнения списков с координатами по оси Х и Y;

ВАРИАНТ СКРИПТА

*Спрайт-ластик*

* По запуску локации происходит задержка величиной не более 0,5 секунды;
* переходит в центр экрана координаты по оси Х – 0, по оси Y – 0;
* Сбрасывает к 0 начальное значение переменной Sn.position, которая управляет переходом к считыванию следующей пары координат спрайта Snake;
* Выполняется цикл перехода в координаты по оси Х и Y, согласно данным из списков координат спрайта Snake;
* Выполняется условие, когда спрайт-Добыча меняет значение переменной Sn.stop на единицу, Спрайт-ластик ждёт и продолжает своё движение, после того, когда значение переменной Sn.stop возвращается к 0;
* При получении сообщения «Начальный экран» – ­­­­­­прячется.

ВАРИАНТ СКРИПТА

*Спрайта-ластика*

*Спрайт-добыча*

* По запуску локации переходит в координаты со случайными значениями по оси Х и Y;
* Сбрасывает к 0 начальное значение переменных: Sn.score и Sn.stop;
* Ждёт касания со спрайтом Snake, после которого увеличивается переменная Sn.score на 1;
* После касания со спрайтом Snake, также происходит увеличение переменная Sn.stop на 1, значение которой после паузы 0,1-0,3 секунды возвращается к 0.

ВАРИАНТ СКРИПТА

*Спрайта-добычи*

**Rocket**

Описание локации:

В игре используется алгоритм движения, когда действующий персонаж перемещается только влево и вправо. Эффект движения вверх создаётся перемещением фона, движение которого направлено вниз. Фоном служит подготовленные спрайты, размеры которых совпадают с размером сцены. Игроку, управляющему *Ракетой,* следует отклоняться от столкновения с препятствием в виде *Планеты*, которая движется ему навстречу. Когда *Ракета* касается *Cолнца,* игра считается пройденной.

Для создания бесконечного цикла смены спрайтов, который создаёт эффект движения вверх необходимо ввести переменную, условно названную ScrollY, которая будет управлять положением каждого из 4-х спрайтов-фонов. При достижении определённого значения переменной её необходимо сбросить на первоначальное значение для повторного исполнения цикла.

Программирование объектов:

*Спрайты-фоны ( Количество – 4шт)*

* по запуску локации перейти в центр экрана и показаться.

ВАРИАНТ СКРИПТА для всех *Спрайтов-фонов*

* Движение спрайтов организуется по следующему алгоритму:

Спрайт 1 Спрайт 2

Спрайт 3 Спрайт 4

*Ракета*

* По запуску локации *Ракета* переходит в начальную точку.

ВАРИАНТ СКРИПТА

* управление перемещением *Ракеты* по оси Х.

ВАРИАНТ СКРИПТА

* Управление переменной ScrollY.

ВАРИАНТ СКРИПТА

* Касание *Планеты -* поражение.
* Касание *Солнца -* победа.

*Планета*

По запуску локации *Планета* ждёт определённое время, затем переходит в координаты по оси Х – случайное значение от -220 до 220; по оси Y – значение 180. Далее движется вниз экрана и исчезает, когда доходит до края экрана.

ВАРИАНТ СКРИПТА

*Солнце*

По запуску локации ждёт от 4 до 10 секунд, затем переходит в координаты по оси Х – случайное значение от – 240 до 240; по оси Y – значение 180. Далее *Солнце* движется вниз экрана со скоростью выше, чем скорость движения *Планеты* и исчезает, когда доходит до края экрана.

ВАРИАНТ СКРИПТА

**Arcanoid**

Описание локации:

Игрок, управляющий перемещением Платформы при помощи указателя мышки, отбивает мяч. Мяч, отскакивающий от стен и Платформы, перемещается по игровому полю, сбивая Блоки, размещенные на поле. Задача игрока состоит в том, чтобы освободить игровое поле от Блоков. В случае касания мяча цвета линии, размещенной ниже Платформы, игра начинается сначала.

Программирование объектов:

*Платформа*

* Управление с помощью указателя мыши;
* По запуску локации *Платформа*  переходит по оси Х в центр экрана, по оси Y вниз экрана;

ВАРИАНТ СКРИПТА

*Мяч*

* По запуску локации переходит в установленные координаты;
* Сбрасывает переменную, считающую количество сбитых *Блоков* к 0;
* Выполняется цикл движения *Мяча* по игровому полю;
* Выполняется условие: при касании *Платформы –* изменить направление;
* Выполняется условие: при касании цвета линии нарисованной ниже *Платформы -*  начать игру сначала;
* Выполняется условие: при достижении счёта 10 очков – переход на начальный экран.

ВАРИАНТ СКРИПТА

*Блок (10 Спрайтов)*

* Каждый блок переходит в соответственные ему координаты;
* Выполняется условие: при касании *Мяча –* прячется и изменяет переменную очки на 1 балл;

ВАРИАНТ СКРИПТА

**Начальный экран\***

На произвольном фоне располагаются три объекта, каждый из которых соответствует одной из мини-игр. При нажатии на объект он передаёт сообщение о запуске одной из мини-игр. При этом отправляется сообщение всем *Спрайтам-объектам* начального экрана, чтобы они спрятались.

Программирование объектов:

*Спрайты начального экрана*

* При нажатии на *Спрайт* он передаёт сообщение о запуске мини-игры:
* Когда каждый из 3-х *Спрайтов* получает сообщение *Игра – Спрайт* исчезает с экрана:

Программирование сцены

Для всех мини-игр используется произвольный фон который, по мнению авторов максимально соответствует игровой локации. Авторы имеют возможность внести изменения в готовый фон из библиотеки приложения Scratch, либо создать в графическом редакторе собственное изображение.

ВАРИАНТ СКРИПТА Сцены

Фон для мини-игры **Ракета** создаётся при помощи алгоритма приведённого в описании локации.

\*Проект считается выполненным, если при запуске программы имеется возможность перейти в любую локацию мини-игры. Переход из одной мини-игры в другую мини-игру не считается обязательным условием выполнения задания.