Министерство цифрового развития, связи и массовых коммуникаций

Российской Федерации Сибирский Государственный Университет

Телекоммуникаций и Информатики СибГУТИ

Кафедра прикладной математики и кибернетики

Расчётно-графическая работа

по дисциплине “Программирование мобильных устройств”

2D игра на движке Unity

Выполнил:

Студент группы ИП-916

Меньщиков Д.А.

Работу проверила:

Павлова У. В.

Новосибирск, 2022

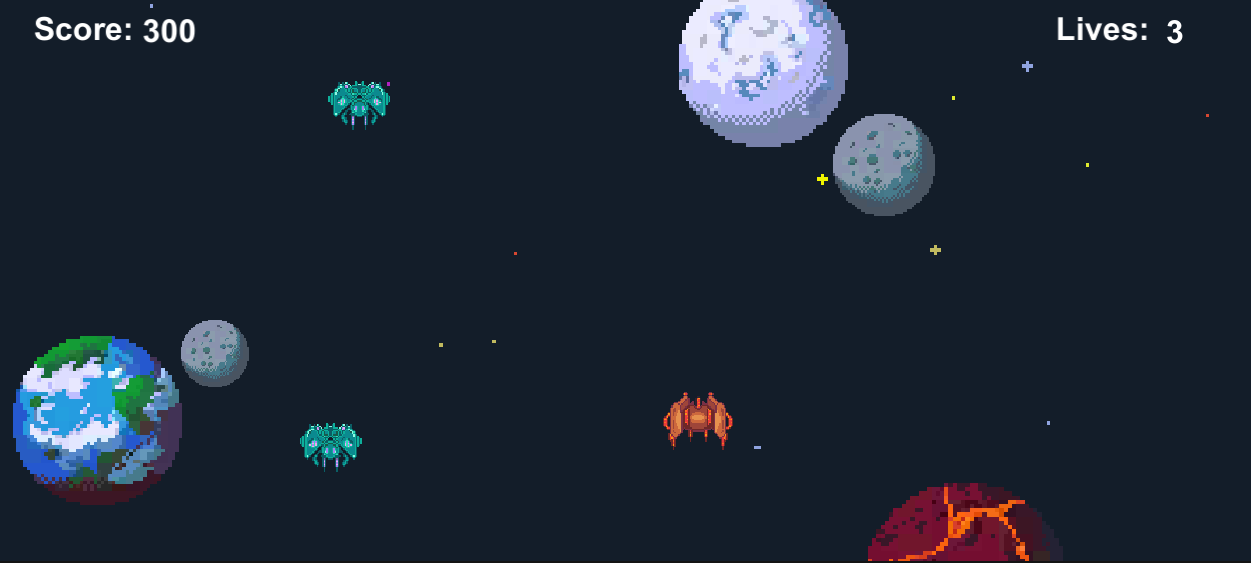
**Задание:**

В рамках выполнения РГР необходимо написать игровое приложение. Сюжет игры выбирается согласно варианту. Сюжет может быть произвольным (по согласованию с преподавателем).

Идеей для игры послужил один из предложенных вариантов сюжетов игры “Звездные врата”, но в игре отсутствует конец, то есть игры из себя представляет бесконечный 2D платформер. Игрок управляет космическим кораблем, ему предстоит уворачиваться от снарядов вражеских кораблей, а так же самому уничтожать вражеские корабли, что бы получить очки. Игра продолжается до тех пор, пока Игрок не потратит все жизни.

**Скриншоты:**

****

****

**Листинг:**

**Управление кораблем**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

using UnityEngine.SceneManagement;

namespace HoloToolkit.Unity

{

    [RequireComponent(typeof(Rigidbody2D))]

    public class SpaceshipMove : MonoBehaviour

    {

        [SerializeField]

        private Joystick joystick;

        private Rigidbody2D rb;

        private Vector2 direction = Vector2.zero;

        private float speed = 5f;

        private const int lives = 3;

        private int currentLives;

        private float time;

        public Text LivesCount;

        public GameObject BeholderBullets;

        public GameObject BulletPosition;

        public float timer = 0f;

        private void Start()

        {

            currentLives = lives;

            LivesCount.text = currentLives.ToString();

            transform.position = Vector2.zero;

            gameObject.SetActive(true);

        }

        private void Awake()

        {

            rb = GetComponent<Rigidbody2D>();

        }

        private void FixedUpdate()

        {

            if (joystick.Horizontal != 0 || joystick.Vertical != 0)

            {

                direction.x = joystick.Horizontal;

                direction.y = joystick.Vertical;

                rb.MovePosition(rb.position + speed \* Time.deltaTime \* direction);

            }

        }

        private void Update()

        {

            Vector2 min = Camera.main.ViewportToWorldPoint(new Vector2(0, 0));

            Vector2 max = Camera.main.ViewportToWorldPoint(new Vector2(1, 1));

            if ((time += Time.deltaTime) > .1f)

            {

                if (Input.GetKeyDown("space"))

                {

                    GameObject bullets = (GameObject)Instantiate(BeholderBullets);

                    bullets.transform.position = BulletPosition.transform.position;

                    time = 0f;

                }

            }

            if (currentLives == 0)

            {

                while(timer < 10f)

                {

                    timer += Time.deltaTime;

                    //print(timer);

                    Debug.Log(timer);

                }

                //SceneManager.LoadScene(SceneManager.GetActiveScene().name);

            }

        }

        private void OnTriggerEnter2D(Collider2D other)

        {

            if((other.tag == "Enemy") || (other.tag == "EnemyBullet"))

            {

                currentLives--;

                LivesCount.text = currentLives.ToString();

                transform.position = Vector2.zero;

                if (currentLives == 0)

                {

                    Destroy(gameObject);

                    SceneManager.LoadScene(SceneManager.GetActiveScene().name);

                }

            }

        }

    }

}

**Движение врагов:**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class EnemyControl : MonoBehaviour

{

    private float speed = 2f;

    GameObject scoreText;

    void Start()

    {

        scoreText = GameObject.FindGameObjectWithTag("Score");

    }

    void Update()

    {

        Vector2 pos = transform.position;

        pos = new Vector2(pos.x + 2f \* Mathf.Sin(pos.y \* 1f) \* Time.deltaTime, pos.y - speed \* Time.deltaTime);

        transform.position = pos;

        Vector2 min = Camera.main.ViewportToWorldPoint(new Vector2(0, 0));

        if(transform.position.y < min.y)

        {

            Destroy(gameObject);

        }

    }

    private void OnTriggerEnter2D(Collider2D collision)

    {

        if (collision.tag == "Bullet")

        {

            scoreText.GetComponent<GameScore>().ScoreCount += 100;

            Destroy(gameObject);

        }

    }

}

**Создание врагов:**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class EnemyCreater : MonoBehaviour

{

    public GameObject Enemy;

    public float maxEnemyCreateInSec = 5f;

    // Start is called before the first frame update

    void Start()

    {

        Invoke("CreateEnemy", maxEnemyCreateInSec);

        InvokeRepeating("IncreaseCreate", 0f, 30f);

    }

    // Update is called once per frame

    void Update()

    {

    }

    private void CreateEnemy()

    {

        Vector2 min = Camera.main.ViewportToWorldPoint(new Vector2(0, 0));

        Vector2 max = Camera.main.ViewportToWorldPoint(new Vector2(1, 1));

        GameObject anEnemy = (GameObject)Instantiate(Enemy);

        anEnemy.transform.position = new Vector2(Random.Range(min.x, max.x), max.y);

        CreateNextEnemy();

    }

    private void CreateNextEnemy()

    {

        float creater = 1f;

        if(maxEnemyCreateInSec > 1f)

        {

            creater = Random.Range(1f, maxEnemyCreateInSec);

        }

        Invoke("CreateEnemy", creater);

    }

    private void IncreaseCreate()

    {

        if(maxEnemyCreateInSec > 1f)

        {

            maxEnemyCreateInSec--;

        }

        if(maxEnemyCreateInSec == 1f)

        {

            CancelInvoke("IncreaseCreate");

        }

    }

}

**Вражеские снаряды:**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

public class EnemyBullets : MonoBehaviour

{

    public float speed;

    Vector2 dir;

    bool isReady;

    void Awake()

    {

        speed = 3f;

        isReady = false;

    }

    void Start()

    {

    }

    public void SetDirection(Vector2 direction)

    {

        dir = direction.normalized;

        isReady = true;

    }

    void Update()

    {

        if(isReady)

        {

            Vector2 position = transform.position;

            position += dir \* speed \* Time.deltaTime;

            transform.position = position;

            Vector2 min = Camera.main.ViewportToWorldPoint(new Vector2(0, 0));

            Vector2 max = Camera.main.ViewportToWorldPoint(new Vector2(1, 1));

            if (transform.position.x < min.x || transform.position.y > max.x || transform.position.y < min.y || transform.position.y > max.y)

            {

                Destroy(gameObject);

            }

        }

    }

    private void OnTriggerEnter2D(Collider2D collision)

    {

        if (collision.tag == "Spaceship" || collision.tag == "Bullet")

        {

            Destroy(gameObject);

        }

    }

}

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class EnemyBulletStart : MonoBehaviour

{

    public GameObject enemyBullet;

    void Start()

    {

        Invoke("Fire", 1f);

    }

    void Update()

    {

    }

    private void Fire()

    {

        GameObject spaceship = GameObject.Find("Beholder");

        if (spaceship != null)

        {

            GameObject bullet = (GameObject)Instantiate(enemyBullet);

            bullet.transform.position = transform.position;

            Vector2 dir = spaceship.transform.position - bullet.transform.position;

            bullet.GetComponent<EnemyBullets>().SetDirection(dir);

        }

    }

}

**Снаряды игрока:**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class SpaceshipShoot : MonoBehaviour

{

    private float speed = 8f;

    void Update()

    {

        Vector2 bulletPos = transform.position;

        bulletPos = new Vector2(bulletPos.x, bulletPos.y + speed \* Time.deltaTime);

        transform.position = bulletPos;

        Vector2 max = Camera.main.ViewportToWorldPoint(new Vector2(1, 1));

        if(transform.position.y > max.y)

        {

            Destroy(gameObject);

        }

    }

    private void OnTriggerEnter2D(Collider2D collision)

    {

        if(collision.tag == "Enemy" || collision.tag == "EnemyBullet")

        {

            Destroy(gameObject);

        }

    }

}

**Создание и движение фона:**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class InfinityBackground : MonoBehaviour

{

    [SerializeField]

    private SpriteRenderer sprite;

    private float speed = 3f;

    private float positionY;

    private Vector2 restartPos;

    private void Awake()

    {

        restartPos = transform.position;

        positionY = sprite.bounds.size.y - restartPos.y;

    }

    // Update is called once per frame

    void Update()

    {

        transform.Translate(Vector3.down \* speed \* Time.deltaTime);

        if(transform.position.y <= - positionY)

        {

            transform.position = restartPos;

        }

    }

}