МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Кафедра вычислительной математики и программирования

**Лабораторная работа №3**

**по курсу «Программирование игр»**

Выполнил: Ваньков Д. А.

Группа: М8О-106М

Преподаватели: Аносова Н.П.

Москва, 2022

### Цель работы

Реализовать 3D игру, в которой космический корабль летит между астероидами.

### Сценарий выполнения работы

### Первоначально были загружены сцена с космосом и префабы корабля с астероидами.

После этого объекты были размещены на сцене и были реализованы скрипты для астероидов и корабля.

MoveAst.cs:

**using** **System.Collections**;

**using** **System.Collections.Generic**;

**using** **UnityEngine**;

**public** **class** **MoveAst** : MonoBehaviour {

**private** **int** count = **0**;

Rigidbody m\_Rigidbody;

**private** **int** o = **0**;

**private** **void** **OnCollisionEnter**(Collision collision) {

**if** (collision.gameObject.tag == "Player") {

**if**(transform.position.x < **0**) {

o = **1**;

} **else** {

o = -**1**;

}

m\_Rigidbody.isKinematic = **false**;

**float** rx = Random.Range(-**4.0f**, -**2.0f**);

**float** ry = Random.Range(-**3.0f**, **3.0f**);

**float** rz = Random.Range(-**3.0f**, **3.0f**);

m\_Rigidbody.AddForce(**new** Vector3(rx \* o, ry, rz) \* **10f**, ForceMode.Impulse);

count++;

}

}

**void** **Start**() {

m\_Rigidbody = GetComponent<Rigidbody>();

}

// Update is called once per frame

**void** **Update**() {

**if**(count == **0**) {

transform.Translate(Vector3.down \* **15f** \* Time.deltaTime);

}

**if**(transform.position.y < -**66.3f**) {

Destroy(gameObject);

}

}

}

Этот скрипт прикреплен к префабу астероида. Скрипт крипт подвергает астероид движению, а когда координата «y» у астероида будет меньше -66.3, астероид удалится со сцены.

Spawn.cs:

**using** **System.Collections**;

**using** **System.Collections.Generic**;

**using** **UnityEngine**;

**public** **class** **Spawn** : MonoBehaviour {

**public** GameObject astr;

**void** **Start**() {

StartCoroutine(create());

}

IEnumerator **create**() {

**while** (**true**) {

**int** randX = Random.Range(-**10**, **11**);

**int** randZ = Random.Range(-**6**, **7**);

**float** rx = (**float**)randX;

**float** rz = (**float**)randZ;

Instantiate(astr, **new** Vector3(rx, -**10f**, rz), Quaternion.identity);

**yield** **return** **new** **WaitForSeconds**(**0.15f**);

}

}

}

Этот скрипт создает астероиды в случайных координатах каждые 0.15 секунд.

Player.cs:

**using** **System.Collections**;

**using** **System.Collections.Generic**;

**using** **UnityEngine**;

**using** **UnityEngine.SceneManagement**;

**using** **UnityEngine.UI**;

**public** **class** **Player** : MonoBehaviour {

**public** **float** speed = **2f**;

**public** **int** CNT = **0**;

Vector3 startpos;

**private** **void** **OnCollisionEnter**(Collision collision) {

**if** (collision.gameObject.tag == "astr") {

transform.position = **new** Vector3(**0.05341797f**, -**56.99992f**, -**0.3f**);

Debug.Log("You hit the asteroid. Total hits: " + CNT);

CNT++;

}

}

**void** **Start**() {

startpos = transform.position;

}

**void** **Update**() {

Vector3 destination = **new** Vector3(-**115f**, **0**, **0**);

**float** vertical = Input.GetAxisRaw("Vertical");

**float** horizontal = Input.GetAxisRaw("Horizontal");

**if** (horizontal < **0**) {

transform.Translate(**new** Vector3(-**1**,**0**, **0**) \* speed \* Time.deltaTime);

} **else** **if** (horizontal > **0**) {

transform.Translate(**new** Vector3 (**1**, **0**, **0**) \* speed \* Time.deltaTime);

} **else** {}

**if** (vertical > **0**) {

transform.Translate(**new** Vector3(**0**, **1**,**0**) \* speed \* Time.deltaTime);

} **else** **if** (vertical < **0**) {

transform.Translate(**new** Vector3(**0**, -**1**, **0**) \* speed \* Time.deltaTime);

} **else** {}

Vector3 go = **new** Vector3(horizontal, vertical, **0**);

}

}

В этом скрипте реализовано перемещение корабля в плоскости камеры, а также счетчик сбитых астероидов, который отображается в DEBUG логе.

### Выводы

В результате выполнения лабораторной работы был реализован симулятор космического корабля. Я более подробно познакомился с взаимодействием объектов в пространстве, а также их перемещении.