МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Кафедра вычислительной математики и программирования

**Лабораторная работа №2**

**по курсу «Программирование игр»**

Выполнил: Ваньков Д. А.

Группа: М8О-106М

Преподаватели: Аносова Н.П.

Москва, 2022

### Цель работы

Реализовать 3D игру, в которой реализовано упругое соударение между шариками, учтен глубинный фактор сцены.

### Сценарий выполнения работы

### Первоначально была создана коробка из 5 примитивов, создающих сцену. Каждый примитив имеет коллайдер, который отвечает за физическую оболочку объекта.

Следующим шагом был создан префаб шарика для игры. Он содержит материал, коллайдер, риджитбоди и физический материал.

После наступил этап написания скриптов. Первый скрипт LMBcs:

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

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

**using** **UnityEngine**;

**public** **class** **LeftButton** : MonoBehaviour {

**public** Camera cameraM;

**public** GameObject obj;

**public** Material[] mat;

**public** Material red;

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

**if** (Input.GetMouseButtonDown(**0**)) {

Ray vec = cameraM.ScreenPointToRay(Input.mousePosition);

RaycastHit hit;

**if** (Physics.Raycast(vec.origin, vec.direction, **out** hit, Mathf.Infinity)) {

**if** (hit.rigidbody == **null** && hit.transform.tag == "plane") {

**int** rand = Random.Range(**0**, mat.Length);

Instantiate(obj, hit.point + **7** \* Vector3.up, Quaternion.identity);

obj.GetComponent<Rigidbody>().useGravity = **false**;

obj.GetComponent<MeshRenderer>().material = mat[rand];

} **else** **if** (hit.transform.tag != "wall") {

hit.transform.gameObject.GetComponent<MeshRenderer>().material = red;

hit.transform.gameObject.tag = "ball";

}

}

}

}

}

По нажатию левой кнопки мыши из местоположения мыши отправляется луч. Если он не сталкивается с риджитбоди, то создается сфера с рандомным цветом. Эта сфера зависает в воздухе, так как гравитация у нее отключена при нажатии на пробел включается гравитация. В противном случае шар окрашивается в красный и получает тег «sh».

Второй скрипт RMB.cs:

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

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

**using** **UnityEngine**;

**public** **class** **RightButton** : MonoBehaviour {

**public** Material[] mat;

Rigidbody rb;

**public** **float** speed;

**public** Material red;

Vector3 go;

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

rb = gameObject.GetComponent<Rigidbody>();

}

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

**if** (collision.gameObject.tag == "ball" || collision.gameObject.tag == "Untagged") {

**if** (gameObject.tag == "ball") {

gameObject.tag = "Untagged";

**int** rand = Random.Range(**0**, mat.Length);

gameObject.GetComponent<MeshRenderer>().material = mat[rand];

}

}

}

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

**if** (rb.useGravity == **false**) {

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

Vector3 go = **new** Vector3(**0**, vert, **0**);

transform.Translate(go.normalized \* **5** \* Time.deltaTime);

**if** (Input.GetKeyDown(KeyCode.Space)) {

rb.useGravity = **true**;

}

}

**if** (Input.GetMouseButtonDown(**1**)) {

**if** (rb.useGravity == **true**) {

Ray vec = Camera.main.ScreenPointToRay(Input.mousePosition);

RaycastHit hit;

**if** (Physics.Raycast(vec.origin, vec.direction, **out** hit, Mathf.Infinity)) {

**if** (hit.rigidbody == **null** && hit.transform.tag == "plane") {

**if** (gameObject.tag == "ball") {

Vector3 t = hit.point - transform.position;

rb.AddForce(t \* speed, ForceMode.Impulse);

}

}

}

}

}

}

}

Если у шарика отключена гравитация мы можем настроить высоту его падения с помощью стрелочек на клавиатуре, а по нажатию клавиши пробел шарик начнет падать. По нажатию правой кнопки мыши все красные шарики с включенной гравитацией отправляются в место клика. При соударении шаров не со стеной каждый красный шар меняет свой цвет на рандомный.

### Выводы

В результате выполнения лабораторной работы была реализована простая 3D игра, основанная на лабораторной работе номер 1. После выполнения данной работы, я познакомился поближе с трехмерным измерением и взаимодействием в нем объектов.