**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ**

***Институт Принтмедиа и информационных технологий***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № 11**

**Дисциплина:** Компьютерная графика

**Тема:** Обводка объектов

**Выполнил:** Вышегородских Виктор Егорович, **студент группы:** 211-728

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия И.О.)

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Проверил: \_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_***

(Фамилия И.О., степень, звание) **(Оценка)**

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва**

**2023**

На основе теоретического материала написали шейдер.

На основе шейдера был создан и настроен материал.

В Unity3D была создана и настроена сцена. Добавили на сцену куб, применили к нему созданный ранее материал на основе доработанного шейдера.

# Исходный код шейдера:

Shader "Unlit/Lab11 1"

{

    Properties

    {

        \_OutlineColor ("Outline Color", Color) = (0, 0, 0, 1)

        \_OutlineThickness ("Outline Thickness", Range(0, 0.1)) = 0.03

        \_Color ("Tint", Color) = (0, 0, 0, 1)

        \_MainTex ("Texture", 2D) = "white" {}

    }

    SubShader

    {

        Tags

        {

            "RenderType" = "Opaque"

            "Queue" = "Geometry"

        }

        LOD 100

        Pass

        {

            CGPROGRAM

            #include "UnityCG.cginc"

            #pragma vertex vert

            #pragma fragment frag

            sampler2D \_MainTex;

            float4 \_MainTex\_ST;

            fixed4 \_Color;

            struct appdata

            {

                float4 vertex : POSITION;

                float2 uv : TEXCOORD0;

            };

            struct v2f

            {

                float4 position : SV\_POSITION;

                float2 uv : TEXCOORD0;

            };

            v2f vert(appdata v)

            {

                v2f o;

                o.position = UnityObjectToClipPos(v.vertex);

                o.uv = TRANSFORM\_TEX(v.uv, \_MainTex);

                return o;

            }

            fixed4 frag(v2f i) : SV\_TARGET

            {

                fixed4 col = tex2D(\_MainTex, i.uv);

                col \*= \_Color;

                return col;

            }

            ENDCG

        }

        Pass

        {

            Cull Front

            CGPROGRAM

            #include "UnityCG.cginc"

            #pragma vertex vert

            #pragma fragment frag

            sampler2D \_MainTex;

            float4 \_MainTex\_ST;

            fixed4 \_Color;

            fixed4 \_OutlineColor;

            float \_OutlineThickness;

            struct appdata

            {

                float4 vertex : POSITION;

                float3 normal : NORMAL;

            };

            struct v2f

            {

                float4 position : SV\_POSITION;

            };

            v2f vert(appdata v)

            {

                v2f o;

                float3 normal = normalize(v.normal);

                float3 outlineOffset = normal \* \_OutlineThickness;

                float3 position = v.vertex + outlineOffset;

                o.position = UnityObjectToClipPos(position);

                return o;

            }

            fixed4 frag(v2f i) : SV\_TARGET

            {

                return \_OutlineColor;

            }

            ENDCG

        }

    }

    FallBack "Standard"

}

# Скриншоты итоговой сцены в игровом движке Unity3D:

Изображение выглядит как текст, монитор, внутренний, экран

Автоматически созданное описание