综合测试题 1 参考答案

一、 选择题（3 分×7 题=21 分）

题号	1	2	3	4	5	6	7
答案	BD	C	AB	AD	ACD	CD	C

二、 判断题（2 分×5 题=10 分）

题号	1	2	3	4	5
答案	√	×	√	×	×

三、 填空题（2 分×8 空=16 分）

1、法向量

2、高洛德着色

3、模板缓存

4、顶点着色器

5、环境光，散射光，镜面光

6、Techinique

四、 名词解释（5 分×2 题=10 分）

交换链：D3D 维护着两个（或以上）纹理缓存，前台缓存用于前台显示，后台缓存用于后台绘制。当前台缓存显示完成后，两个缓存交换位置，前台缓存变成后台缓存，后台缓存变成前台缓存。这个过程称为交换链。

混合：混合是指将两个颜色应用不同的混合公式产生一些特殊效果（比如半透明效果），除此之外还可以用于模拟多重纹理的效果。

五、 简答题（答中要点即可）（7 分×3 题=21 分）

1、初始化 Direct3D 的主要步骤：

① 填充 DXGI_SWAP_CHAIN_DESC 结构；

② 使用 D3DCreateDeviceAndSwapChain 函数和 1 中填充好的 DXGI_SWAP_CHAIN_DESC 结构来创建 ID3D11Device、IDXGISwapChain 和 ID3D11DeviceContext；

③ 创建目标渲染观察视图 ID3D11RenderTargetView 并将其绑定到渲染管线；

④ 创建视口（View Port）。

2、Effect文件由以下不同的部分组成。

外部变量：存储从应用程序得到的数据的变量

输入输出结构：能够在着色器之间传递的结构。例如顶点着色器输出的信息会被传递给像素着色器，作为像素着色器的输入。

顶点着色器：处理顶点的部分

像素着色器：处理像素的部分

Technique：定义 Effect 文件中使用的通道（Pass），通道是定义 Effect 中所使用的着色器和一些状态。 一个 Technique 可以有多个 Pass。下面就介绍如何编写 SimpleShader.fx 文件。

3、 输入装配阶段

顶点着色器阶段

外壳着色器阶段

曲面细分阶段

域着色阶段

几何着色阶段

光栅化阶段

像素着色阶段

输出合并阶段

六、 综合题（共 22 分）

1、int main()

```
{
    //声明3个XMMATRIX对象，
    //分别用来表示平移矩阵（mTrans）,旋转矩阵（mRota）,以及缩放矩阵（mScal）
    XMMATRIX mTrans, mRota, mScal;
    //生成缩放矩阵
    mScal = XMMatrixScaling(5.0f, 5.0f, 5.0f);
    //生成旋转矩阵
    mRota = XMMatrixRotationY(XM_PIDIV4);
    //生成平移矩阵
    mTrans = XMMatrixTranslation(0.0f, 0.0f, -4.0f);
    //将上面生成的3个变换矩阵组合成一个最终的变换矩阵
    XMMATRIX mFinal;
    //所以这里首先将mScal和mRota相乘的中间结果放入mFinal中
    mFinal = XMMatrixMultiply(mRota, mTrans);
    //再将中间结果与mTrans相乘，得到最终结果并覆盖先前的mFinal
```

```
    mFinal = XMMatrixMultiply(mFinal, mScal);

    //声明一个XMVECTOR对象
    XMVECTOR vector= XMVectorSet(1.0f, 2.0f, 5.0f, 1.0f);

    //将上面生成的最终变换矩阵应用到XMVECTOR对象上
    //并将生成的新向量覆盖原来的向量
    vector = XMVector4Transform(vector, mFinal);
    return 0;

}
```

2、（1）填充 D3D11_BLEND_DESC 结构

（2）利用填充好的D3D11_BLEND_DESC结构创建ID3D11BlendState 对象

（3）在绘制物体之前，调用immediateContext->OMSetBlendState(blendStateAlpha, BlendFactor, 0xffffffff);
将混合状态设置到设备上下文

（4）绘制物体之后，调用immediateContext->OMSetBlendState(0,0,0xffffffff);关闭混合状态

```
//先创建一个混合状态的描述
    D3D11_BLEND_DESC blendDesc;
    ZeroMemory(&blendDesc, sizeof(blendDesc)); //清零操作
    blendDesc.AlphaToCoverageEnable = false;    //关闭AlphaToCoverage多重采样技术
    blendDesc.IndependentBlendEnable = false;       //不针对多个RenderTarget使用不同的混合状态
    //只针对RenderTarget[0]设置绘制混合状态，忽略1-7
    blendDesc.RenderTarget[0].BlendEnable = true;                           //开启混合
    blendDesc.RenderTarget[0].SrcBlend = D3D11_BLEND_SRC_ALPHA;         //设置源因子
    blendDesc.RenderTarget[0].DestBlend = D3D11_BLEND_INV_SRC_ALPHA;//设置目标因子
    blendDesc.RenderTarget[0].BlendOp = D3D11_BLEND_OP_ADD;                 //混合操作
    blendDesc.RenderTarget[0].SrcBlendAlpha = D3D11_BLEND_ONE;          //源混合百分比因子
    blendDesc.RenderTarget[0].DestBlendAlpha = D3D11_BLEND_ZERO;        //目标混合百分比因子
    blendDesc.RenderTarget[0].BlendOpAlpha = D3D11_BLEND_OP_ADD;        //混合百分比
```

的操作

```
    blendDesc.RenderTarget[0].RenderTargetWriteMask =
D3D11_COLOR_WRITE_ENABLE_ALL;   //写掩码
    //创建ID3D11BlendState接口
     device->CreateBlendState(&blendDesc, &blendStateAlpha);
```