



İsim		No	
------	--	----	--

1.

```
D3DXMatrixRotationY( &mRotateY, -t );  
D3DXMatrixTranslation( &mTranslate, -4, 3, 0 );  
D3DXMatrixScaling( &mScale, 0.2f, 0.2f, 0.2f );  
g_World = mScale * mRotateY * mTranslate * mRotateY;
```

a) Yukarıdaki kod parçası ile hesaplanan g_World matrisinin herhangi bir cisme hangi transformasyonu yaptırdığını sözel olarak ifade ediniz. (10P)

CEVAP :	

b) D3DXMatrixTranslation(&mTranslate, -4,3,0); Yukarıdaki satır (4,3,0) 'dan başka hangi şekillerde olsaydı sonuç değişmezdi? İpucu → İki alternatif var.(10P)

CEVAP :

SimpleVertex vertices[] =

```
{  
  { D3DXVECTOR3( 0, 2, 1), D3DXVECTOR2(?, ?) },  
  { D3DXVECTOR3( 1, 0, 1), D3DXVECTOR2(?, ?) },  
  { D3DXVECTOR3( 1, 0, -1), D3DXVECTOR2(?, ?) },  
  { D3DXVECTOR3( 0, 2, -1), D3DXVECTOR2(?, ?) },  
  
  { D3DXVECTOR3( 0, -2, 1), D3DXVECTOR2(?, ?) },  
  { D3DXVECTOR3( 0, -2, -1), D3DXVECTOR2(?, ?) },  
  { D3DXVECTOR3( 1, 0, -1), D3DXVECTOR2(?, ?) },  
  { D3DXVECTOR3( 1, 0, 1), D3DXVECTOR2(?, ?) }  
};
```

DWORD indices[] =

```
{  
  0,1,2,    0,2,3,    4,5,6,    4,6,7  
};
```

2. Yukarıdaki kod parçasında ? ile gösterilen yerler için gerekli doku koordinatlarını aşağıya yazınız. (20P)

CEVAP :

```
D3DXVECTOR2( , )  
D3DXVECTOR2( , )  
D3DXVECTOR2( , )  
D3DXVECTOR2( , )
```

```
D3DXVECTOR2( , )  
D3DXVECTOR2( , )  
D3DXVECTOR2( , )  
D3DXVECTOR2( , )
```

3.

```
for(int i=0; i<3; i++)  
{  
  toLight = normalize( vLightPos[i] - input.PosW );  
  
  dotEyeNorm = dot( toLight, input.Norm );  
  if( dotEyeNorm > 0.0F )  
    diffuseColor = dotEyeNorm * vLightColor[i] ;  
  else  
    diffuseColor = float3( 0.0F, 0.0F, 0.0F );  
  
  toEye = normalize( EyePos - input.PosW );  
  reflected = toLight - 2 *  
    dot(toLight, input.Norm) * input.Norm;  
  
  dotEyereflected = dot( toEye, reflected );  
  if( dotEyereflected > 0.0F )  
    specularColor = pow( dotEyereflected, 64.0F )  
      * vLightColor[i] ;  
  else  
    specularColor = float3( 0.0F, 0.0F, 0.0F );  
  
  finalColor.rgb += saturate( 0.5F * diffuseColor  
    + 0.5F * specularColor );  
}
```

Diffuse ve specular renk bileşenlerini hesaplayan yukarıdaki kod parçası problemsiz derlenmesine rağmen ekranda hatalı bir görüntü çizilmektedir. Programda gerekli değişiklikleri yaparak hatayı düzeltiniz. (20P)

CEVAP :	

4. .cpp uzantılı DirectX10 programında setlenen g_World matrisinin değeri .fx uzantılı HLSL programındaki World matrisine hangi fonksiyonların içinde hangi emirler koşularak aktarılır? (20P)

CEVAP :	

5. g_pd3DDevice ve g_pSwapChain nesneleri backbuffer ile ilgili hangi işlemleri hangi emirlerle gerçekleştirebilirler? (20P)

CEVAP :	