



Sınavda Uyulması Gereken Kurallar

1. Cep telefonlarının, hesap makinesi, saate bakmak gibi herhangi bir amaçla kullanılması **yasaktır**. Telefon kapalı ve cepte olmalıdır.
2. Sorularda anlaşılmayan bir nokta varsa ilk **15 dakika** sorulabilir. Her öğrencinin **1** kez soru sorma hakkı vardır.

1. Köşe noktalarının koordinatları **W0, W1, W2** olarak verilen üçgen içinde **W0** noktasından (**16, 24, 120**) noktasına gitmek için gerekli barisentrik koordinatları hesaplayınız. **(20 PUAN)**

$$W0 = (-40, 40, 120)$$

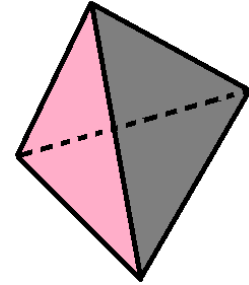
$$W1 = (40, 40, 120)$$

$$W2 = (-40, -40, 120)$$

- 2.

```
D3DXVECTOR3 vertices[] =
{
    D3DXVECTOR3( -1.0f,  0.0f,  1.0f ),
    D3DXVECTOR3(  1.0f,  0.0f,  1.0f ),
    D3DXVECTOR3(  0.0f,  0.0f, -1.0f ),
    D3DXVECTOR3(  0.0f,  1.0f,  0.0f ),
};

DWORD indices[] =
{
    ...
}
```



Şekil 1: Düzgün Dörtüzlü

Yukarıdaki `vertices[]` isimli vertex bufferi kullanarak Şekil 1 'deki gibi bir düzgün dörtüzlü (tetrahedron) çizmek için `indices[]` isimli index buffer nasıl setlenmelidir? **(30 PUAN)**

İpucu → Index bufferda 4 tane üçgen tanımlayacaksınız.

3.

```
D3DXMatrixRotationY( &mRotate, -t );
D3DXMatrixTranslation( &mTranslate1, -2.0f, 0.0f, 0.0f );
D3DXMatrixTranslation( &mTranslate2, -4.0f, 0.0f, 0.0f );
```

Yukarıda tanımlanmış `mRotate`, `mTranslate1`, `mTranslate2` matrislerini kullanarak herhangi bir cismin $(-4.0f, 0.0f, 0.0f)$ noktası etrafında 2 birim yarıçaplı daire çizecek şekilde dönmesi için gerekli `g_World` transformasyon matrisini hesaplayınız. **(30 PUAN)**

4. Aşağıdaki `vertices[]` isimli vertex bufferda **D3DXVECTOR2(?, ?)** ile gösterilen yerlere gerekli doku (texture) koordinatlarını yazınız. **(20 PUAN)**

```
SimpleVertex vertices[] =
{
    { D3DXVECTOR3( -1.0f, -1.0f, -1.0f ), D3DXVECTOR2( ?, ? ) },
    { D3DXVECTOR3(  1.0f, -1.0f, -1.0f ), D3DXVECTOR2( ?, ? ) },
    { D3DXVECTOR3(  1.0f,  1.0f, -1.0f ), D3DXVECTOR2( ?, ? ) },
    { D3DXVECTOR3( -1.0f,  1.0f, -1.0f ), D3DXVECTOR2( ?, ? ) },
};
```