



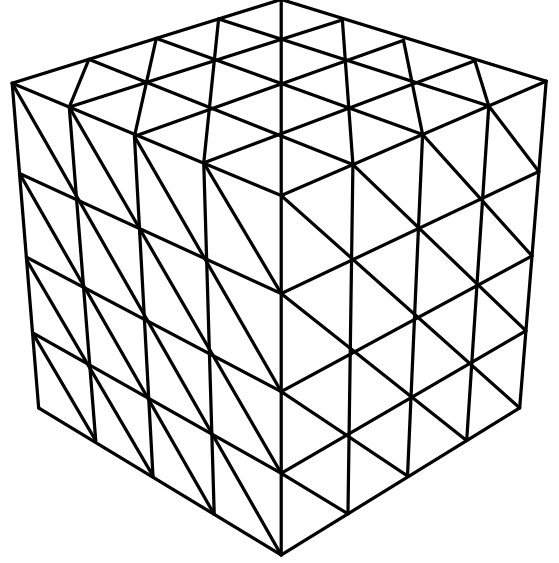
CEVAPLAR

1) DirectX 12'de **yalnızca integer değerler tutan** buffer'lara iki örnek veriniz. Buffer'ların bu integer değerleri ne amaçla tuttuklarını kısaca açıklayınız. (30P)

Örnek olarak Index Buffer ve Stencil Buffer verilebilir.

Index Buffer : Vertex Buffer'daki vertexlerin indisleri 3'erli gruplar halinde düzenlenerek Üçgenler tanımlanır.

Stencil Buffer : 0-255 arası integer değerler alır. Back buffer ile aynı çözünürlüktedir. Aynasal cisimler Back Buffer'a çizilirken aynı piksellerin Stencil Buffer'daki eşleniklerine Stencil Reference denen özel bir sayı yazılarak o pikseller ayna olarak setlenir. Sonra ortamdaki başka cisimlerin aynasal cisimler üzerine yansımaları çizilirken bu Stencil Reference değişkenine bakılır. Yani yansımalar yalnızca Stencil Buffer'da ayna olarak setlenen piksellere karşılık gelen Back Buffer piksellerine çizilir.



```
struct Vertex
{
    XMFLOAT3 position;
    XMFLOAT4 color;
};
Vertex vertices[] = {...}
DWORD indices[] = {...}
```

2) Şekilde verilen küpler topluluğu :

- Yalnızca **vertex buffer** ile (**Vertex**'lerden üçgenler üretilerek) çizildiğinde vertex bufferın **byte** cinsinden boyu **X** olsun.
- **vertex buffer + index buffer** ile (**Vertex**'ler vertex buffera 1'er kez yazılıp, üçgenler index bufferda tanımlanarak) çizildiğinde vertex ve index bufferların **byte** cinsinden toplam boyu **Y** olsun.

X ile Y arasındaki fark nedir?

(10P)

A) 11000 B) 11040 **C) 11080** D) 11120 E) 11160

XMFLOAT3 veri tipi 3 tane **FLOAT** sayı içerir.
FLOAT: 4byte, **DWORD**: 4byte alınız.

X nedir? → **16128**

(15P)

Y nedir? → **5048**

(15P)

98 vertices, 192 triangles
193*3=576 576*28=**16128** (Only Vertex Buffer)
98*28=2744 576*4=2304
2744+2304=**5048** (Vertex Buffer+Index Buffer)

```
Rot45 = XMMatrixRotationY(XM_PI / 4); // 45 CW  
Trans = XMMatrixTranslation(8.0f, 0.0f, 0.0f);  
Scale = XMMatrixScaling(0.5f, 0.5f, 0.5f);
```

```
g_World_1 = Trans * Scale * Rot45 * Scale * Rot45;  
g_World_2 = Trans * Rot45 * Scale * Rot45 * Scale;
```

```
g_World_3 = Scale * Trans * Rot45 * Scale * Rot45;  
g_World_4 = Rot45 * Trans * Scale * Rot45 * Scale;
```

3) Merkezi (0,0,0) olan büyük küpe uygulanan yukarıdaki $g_world_#$ transformasyon matrislerinden **hangi ikisi birbirinin eşdeğeri**? (15P)

g_World_1 & g_World_2

İki eşdeğer transformasyona ait küpü yuvarlak içine alınız. (15P)

