



CEVAPLAR

$V_0(40, 35, 40)$ $V_1(100, 35, 120)$ $V_2(40, -25, 40)$

1) Yukarıda köşe noktaları verilen V üçgeni üzerindeki $iPoint(82, 23, 96)$ noktasının (u, v) barisentrik koordinatlarını hesaplayınız. (30P)

```
L = iPoint - V0 = (42, -12, 56)
Rdu = (V1-V0).Normalize() = (0.6, 0, 0.8)
Su = L * Rdu = 25.2 + 44.8 = 70
u = Su / (V1-V0).Length() = 70 / 100 = 0.7

Rdv = (V2-V0).Normalize() = (0, -1, 0)
Sv = L * Rdv = 12
v = Sv / (V2-V0).Length() = 12 / 60 = 0.2
```

```
void OnRender()
{
    .
    .
    m_commandList->DrawInstanced(3, 1, 0, 0);
    .
    .
    // Present the frame.
    .....
}
```

2) DirectX 12'de çizim işlemlerine ait kodları içeren `OnRender()` fonksiyonunun bir kısmı yukarı paylaşılmıştır. (30P)

a) `DrawInstanced()` fonksiyonu hangi buffer'a çizim yapmaktadır? → [BACK BUFFER]

b) İlgili buffer'a çizim işlemi bittiğinde o buffer'ın içeriği ekranda hangi fonksiyonla paylaşılır? → [PRESENT()]

c) Çizim yapılan buffer'ın içeriğinin ekranda paylaşılmasını hangi nesne yapar? → [SWAP CHAIN]

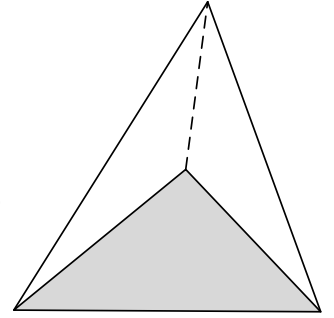
3) STL dosya türünün OBJ dosya türüne kıyasla eksik olan tarafı nedir? (15P)

OBJ dosya türü doku koordinatları içerebilir. STL dosya türünde doku koordinatları yoktur.

```
struct Vertex
{
    XMFLOAT3 position;
    XMFLOAT4 color;
};

Vertex vertices[] = {...}

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



4) Bir Düzgün Dört Yüzlü (Tetrahedron):

- 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? (25P)

Not → **XMFLOAT3** veri tipi 3 tane **FLOAT** sayı içerir.

FLOAT: 4byte, DWORD: 4byte alınız.

A) 152 B) 160 C) 168 **(D) 176** E) 184

$$X = 12 * 7 * 4 = 336$$

$$Y = 4 * 7 * 4 + 12 * 4 = 160$$

$$336 - 160 = 176$$