

1. What effects are lost in Phong illumination when the light source and the viewer infinitely far away from the scene?
2. Discuss two effects that are missed by Phong illumination and interpolation that are captured by ray-tracing.
3. Describe the Z-buffer algorithm. For what type of scenes Z-buffer does not perform well?
4. Describe the role of the focal length for perspective projection in the pin hole camera model.
5. Describe one approach for computing shadows.
6. Explain what a uniform cubic B-spline curve is, and what basis functions are. You may simply state the important properties and sketch a plot.
7. Explain how to compute piecewise-linear approximation to the spline curve using subdivision.
8. Complete the comparison table of cubic curve models.

	Hermite	Bezier	Uniform B-Spline	Nonuniform B-Spline
Convex hull defined by control points				
Perspective transformation invariant				
Continuities inherent in representation				
Number of parameters controlling a curve segment				

9. Define complementary colors and dominant colors using the CIE color model.
10. What is the mipmap texture? Describe the pros and cons of it.
11. Answer True or False.
 - A. The shading computed by Phong shading is C1 continuous.
 - B. Back-face culling is useful for ray-casting.
 - C. A-buffer can handle the transparency.
 - D. Bump mapping changes the geometric shape of the bump mapped surface.
 - E. The BSP tree for a scene is computed in a viewpoint independent manner.
 - F. The performance of a BSP tree representation is not influenced by the selection of splitting planes.
12. What is the fractal dimension of Koch curve? Explain the meaning of this dimension.

13. Explain 5 heuristics among Nielsen's 10 heuristics for user interface design.

14. [Direct X programming]

- A. What kind of a vertex is declared by following FVF? Describe all elements and their dimensions.

```
D3DFVF_XYZ | D3DFVF_TEX3 | D3DFVF_TEXCOORDSIZE3(1)
```

- B. Build a C/C++ structure which is suitable for FVF in A. You can use either C/C++ primitive types or D3DX types such as D3DXVECTOR3.

15. [Direct X programming] Following code is a part of Direct3D application. Assume that pDev, pVB is properly generated D3D device and vertex buffer object, respectively.

```
#define D3DFVF_SAMPLEVERTEX (D3DFVF_XYZ| D3DFVF_DIFFUSE)
struct SAMPLEVERTEX
{
    D3DXVECTOR3 position;
    D3DCOLOR diffuse;
};

SAMPLEVERTEX myVertices[] = // store vertex data in CPU memory
{
    { D3DXVECTOR3( 0.0f,10.0f, 0.0f), 0xFFFFFFFF},
    { D3DXVECTOR3(10.0f,10.0f, 0.0f), 0xFFFFFFFF},
    { D3DXVECTOR3( 0.0f, 0.0f, 0.0f), 0xFFFFFFFF},
    { D3DXVECTOR3(10.0f, 0.0f, 0.0f), 0xFFFFFFFF},
    { D3DXVECTOR3( 0.0f,-10.0f, 0.0f), 0xFFFFFFFF},
    { D3DXVECTOR3(10.0f,-10.0f, 0.0f), 0xFFFFFFFF}
};

int Render(void)
{
    SAMPLEVERTEX* pVertices;
    if( FAILED( pVB->Lock( 0, 0, (void**)&pVertices, 0 ) ) ) return E_FAIL;
    memcpy(pVertices, myVertices, sizeof(SAMPLEVERTEX)*6);
    pVB->Unlock();

    pDev->SetRenderState(D3DRS_SHADEMODE, D3DSHADE_PHONG);
    pDev->SetStreamSource( 0, pVB, 0, sizeof(SAMPLEVERTEX));
    pDev->SetFVF( D3DFVF_SAMPLEVERTEX );

    pDev->BeginScene();
    pDev->DrawPrimitive( D3DPT_TRIANGLESTRIP, 0, 4);
    pDev->EndScene();
    pDev->Present(NULL, NULL, NULL, NULL);
    return S_OK;
}
```

- A. There is a D3D constant which lead to undefined behavior in this code. What is it? Describe why.
- B. Correcting the flaw of A, what is the result polygon which this routine renders? Describe the shape of entire polygon and the topology of triangles.