

- 1- Back Face: os triangulos pertencentes a faces voltadas de costas para a câmara não são desenhados (GPU=>+rapido)
Occlusion: os polígonos que estejam tapados por outros poligonos não são desenhados(GPU=>+complexo, necessario conhecer a restante geometria)

Ficha de Consolidação V

Culling

1. Compare em termos computacionais os três tipos de culling apresentados na disciplina.
2. Descreva o processo matemático para obter a equação normalizada do plano que contém os pontos p_1 , p_2 e p_3 .
3. Descreva os passos necessários para implementar o algoritmo de View Frustum Culling.
4. Considere os vectores d e up , o ponto p , e as distâncias $farDist$, $Wfar$ e $Hfar$, apresentados na figura. Descreva o processo matemático para obter o ponto fbr .

1. calcular os planos da caixa

2. para cada polígono verificar se se encontra dentro ou fora da caixa

Metodos para detetar colisoes:

AABB, OBB, Sphere, Convex Hull

```
c = P+ farDist*d
c' = c + (-up * Hfar/2)
fbr = c' + (d*up) * (Wfar/2)
```

$$\tan(\text{fovy}/2) = \text{Hfar}/(2 * \text{farDist})$$
$$\Leftrightarrow \text{Hfar} = 2 * \tan(\text{fovy}/2) * \text{farDist}$$

$$W_{far} = \text{ratio} * H_{far}$$

$$W_{far} = ratio * 2 * \tan(fovy) * farDist$$

$v = fbr - c$
 $u = c' - c$
 $N = u^*v / ||u^*v||$
 $eq: N_x * x + N_y * y + N_z * z + d = 0$
 substituir por um ponto do plano
 para obter valor de d

5. Considere agora que tem somente os dados presentes nas seguintes instruções:

```
gluPerspective(fov, ratio, nearDist, farDist);
gluLookAt(px,py,pz, lx,ly,lz, ux,uy,uz);
```

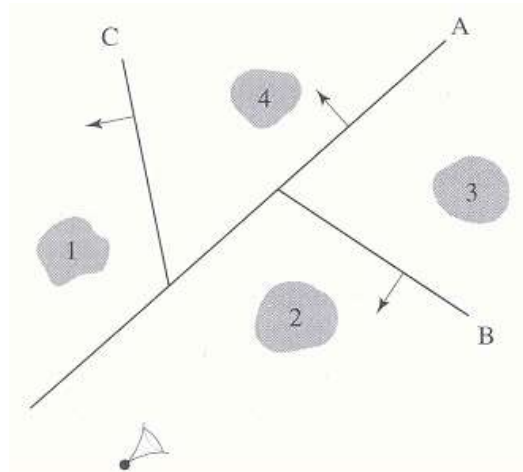
Descreva o processo matemático para obter os dados referidos na pergunta anterior: vectores d , up e $right$, e as distâncias $Wfar$ e $Hfar$.

6. Apresente o algoritmo para extrair os planos do view frustum segundo a visão geométrica.
7. Descreva o algoritmo para extrair os planos do view frustum em clip space.
8. Por forma a tornar eficiente o algoritmo de view frustum culling é necessário implementar algum mecanismo de agrupamento de triângulos. Descreva o processo de partição espacial baseado em k-D trees. Este algoritmo consiste no corte das arestas do triângulo pelo plano do frustum.

Este algoritmo consiste na perda dos graus de liberdade na escolha dos planos para realizar a partição. Escolhe-se um eixo perpendicular a um plano, dividindo o espaço em dois iterativamente.

10. Se existirem triângulos que são divididos por planos, uma hipótese seria fazer cópias dos triângulos em cada um dos espaços (aumento nos requisitos de armazenamento e processamento). O triângulo seria atribuído ao espaço que contém o seu ponto central (perda de detalhes ou precisão). Adaptar a partição de modo a acomodar triângulos que pertencem ao mesmo espaço/pai (complexidade adicional ao algoritmo, lidar com a sobreposição e a interação entre os filhos)

9. Os processos de partição espacial são em regra recursivos na construção da estrutura de dados. Indique três critérios possíveis para terminar a recursividade.
10. Num processo de partição espacial é possível que um triângulo pertença a mais que um filho. Indique quais as opções disponíveis nestes casos apresentando as vantagens e desvantagens de cada uma.
11. Considere a seguinte divisão do espaço utilizando uma BSP. Construa a árvore correspondente e, dada a posição da câmara indicada na figura, apresente a ordem de desenho dos objectos de forma a minimizar a escrita de pixels.



12. Indique os tipos de volumes envolventes que poderiam ser utilizados numa partição hierárquica, comparando a sua eficiência em termos de culling e complexidade algorítmica.
13. Descreva detalhadamente o processo otimizado de teste de inclusão no VFC com paralelepípedos alinhados com os eixos.