

# Aula 6

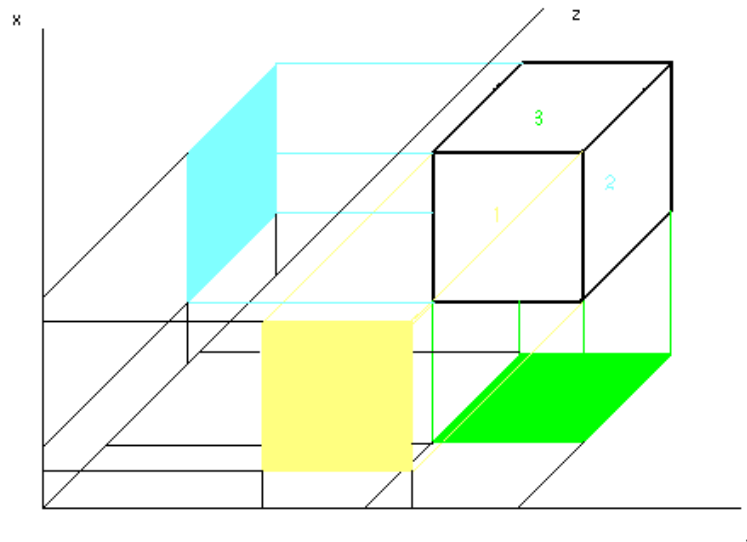
Professora

**Drucilla do Bem Oliveira**

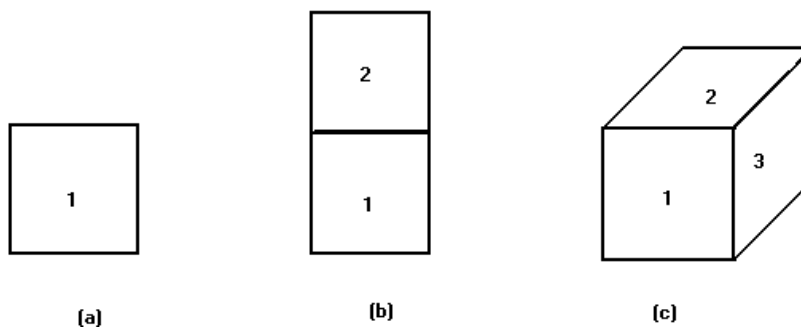
[drucoliveira@hotmail.com](mailto:drucoliveira@hotmail.com)

## HIDDEN LINES

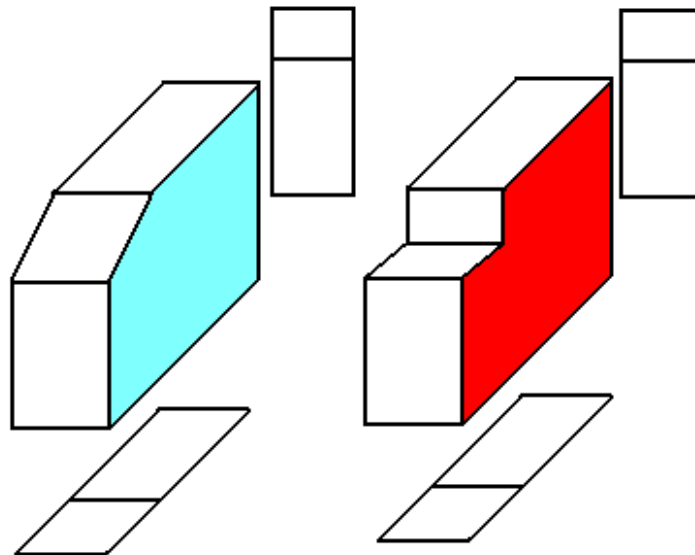
- Os objetos podem ser representados graficamente de duas formas:
  - **a** - no plano, ou
  - **b** - no espaço.
- No espaço esta representação se faz por perspectiva, e no plano mediante projeção do objeto.
- Em desenho técnico normalmente se usa a "soma" das projeções para representação do objeto no espaço (perspectiva), quando ainda não disponível nesta forma.



- Há que se estar atento aos objetos quando em perspectiva, visto que nesta condição nem todas as suas faces estarão visíveis.
- Por exemplo, sabemos que um cubo tem seis faces, mas dependendo do ponto do qual se olha (o chamado ponto de vista ou de observação), este poderá ter uma, duas ou, no máximo, três faces visíveis, como mostra a figura 2.



- Hidden line é prática padrão para representar qualquer linha ou aresta de um objeto que está escondida da visão, levando em conta o ponto de vista de que o objeto é observado.
- Um desenhista decidindo se uma linha deve ser representada como visível ou não, confia nas projeções de contorno do objeto, onde se deve observar a apresentação no plano de, no mínimo, três diferentes vistas do objeto.
- Vista de frente, superior, e lateral.
- Esta exigência se deve ao fato da possibilidade de ocorrência de duas vistas serem iguais para objetos diferentes, onde a particularidade de cada um somente pode ser percebida com o auxílio de mais uma vista, por exemplo à vista da face em cor na fig. 3.



## Formas de apresentação de Hidden lines

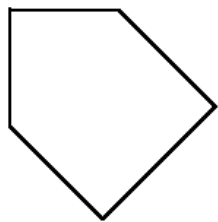
- Em desenho técnico as linhas de contorno do objeto são representadas nas projeções por linhas cheias ou tracejadas, dependendo se visíveis ou não, respectivamente.
- Em perspectiva há três formas de se representar às linhas não visíveis, que são:
  - **a** – Tracejadas, de mesma cor das visíveis;
  - **b** – Tracejadas, com cor diferente das visíveis; ou
  - **c** - . Simplesmente não são traçadas.
- Já para superfícies, normalmente não são traçadas as linhas não visíveis, visto que o traçado destas pode prejudicar a visualização do objeto, produto final.

## Definição de linhas visíveis e invisíveis

- Em desenho técnico são consideradas visíveis todas as linhas de contorno que se enxerga a partir de um determinado ponto de vista, ou seja, não há barreiras entre o observador e a linha de contorno do objeto.
- Ou ainda, as linhas visíveis são aquelas que apresentam a menor distância relativa entre objeto e observador, considerando o ponto de vista.
- Já as linhas invisíveis são àquelas que se encontram por trás de uma face ou de qualquer obstáculo do próprio objeto.
- Deverão se traçar todas as linhas invisíveis do objeto quando este estiver sendo desenhado no plano (em projeção), de forma a facilitar a obtenção da perspectiva.
- Já no espaço não se faz necessário, salvo quando o objeto apresentar detalhes que o exijam.
- Em computação gráfica estes conceitos são muito importantes, principalmente, porque para o computador não faz diferença se as arestas ou superfícies são visíveis ou não.
- Há então, que se elaborar uma forma de identificação de linhas, arestas ou superfícies visíveis e não visíveis, principalmente para que se possa decidir se a mesma deverá ser traçada ou não, visando principalmente fidelidade e resolução.

## Teste de visibilidade

- Considerando um único objeto, este convexo, ou seja, em qualquer posição que se trace um segmento de reta haverá interseção em no máximo dois pontos distintos do objeto.
- Caso contrário, havendo interseção em mais de dois pontos, o objeto é chamado de não convexo.
- Ou ainda, um segmento de reta estará sempre dentro do objeto, o que não acontece com o não convexo.

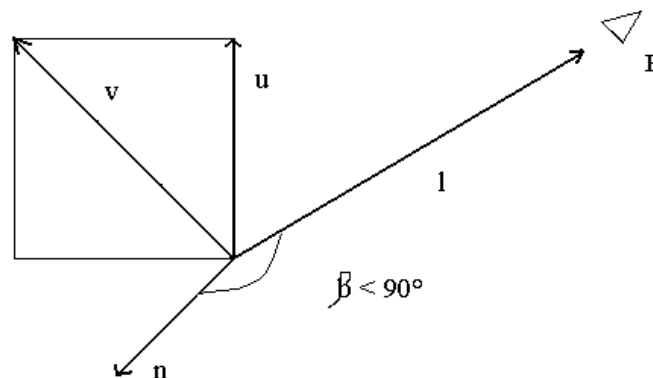


**Polígono convexo**



**Polígono não convexo**

- Para realização do teste de visibilidade haveremos de considerar dois vetores de orientação associados a cada uma das faces ou superfícies,
  - O vetor normal ( $n$ ) a cada uma destas faces ou
  - Superfícies e o vetor da linha de visibilidade ( $l$ ).
- A verificação de visibilidade é então realizada mediante cálculo do ângulo ( $\beta$ ) entre o vetor normal ( $n$ ) de cada uma das faces do objeto em relação ao vetor da linha de visibilidade ( $l$ ), que é o vetor que se encontra sobre a reta suporte desde o olho do observador até o ponto de encontro de dois vetores ( $u$  e  $v$ ) localizados sobre a face em estudo.
- Vetor normal  $n$  é aquele perpendicular a face em estudo, calculado através do produto vetorial ( $u \times v$ ) de dois vetores contidos na face do objeto.
- De álgebra linear, sabemos que o *produto escalar* relaciona dois vetores e o ângulo entre eles, e que tem a seguinte propriedade:
  - $n \cdot l = |n| \cdot |l| \cdot \cos \beta$ ,
  - sendo então:
  - onde:  $|n|$  corresponde a magnitude do vetor normal à face;
  - $|l|$  corresponde à magnitude do vetor  $l$ , que é igual à distância da face ao olho do
  - observador; e
  - $\beta$  o ângulo entre os vetores  $n$  e  $l$ .
- Se o ângulo ( $\beta$ ) entre o vetor normal ( $n$ ) e o vetor da linha de visibilidade ( $l$ ) se encontrar entre  $0^\circ$  e  $90^\circ$  e  $n \cdot l > 0$ , a superfície é visível e pode ser, traçada, disponibilizada na tela ou impressa.
- Se o ângulo ( $\beta$ ) estiver entre  $90^\circ$  e  $180^\circ$  e  $n \cdot l < 0$ , a superfície estará por trás de alguma outra e portanto invisível e indisponível para apresentação ou impressão, ou caso queira exibí-la esta será diferente das visíveis, por exemplo, tracejada.



**Vetores de orientação ( $u$  e  $v$ ), vetor normal ( $n$ ) e de visibilidade ( $l$ )  $\beta < 90^\circ$ .**

## Conclusão

- A partir destas informações básicas é possível identificar as faces visíveis e não visíveis de um objeto, considerando o ponto de vista, e tomar decisões quanto a traçá-las ou não.
- Em computação gráfica isto é feito através de softwares, que deverão realizar as seguintes tarefas:
- Ler as coordenadas do objeto no espaço tridimensional, considerando um ponto de referência e armazená-las em forma de matriz.
- - Localizar no espaço a posição do observador, através do qual definirá os parâmetros de visibilidade.
- - Calcular o vetor normal de cada face do objeto
- - Calcular o vetor da linha de visibilidade para cada face do objeto
- - Realizar o teste de visibilidade calculando o produto escalar entre os dois vetores:
- Se  $n \cdot l > 0$ , a face estará visível
- Se  $n \cdot l < 0$ , a face estará invisível
- - Definir os cantos das faces do objeto e armazená-los de forma matricial
- - Verificar os cantos visíveis, com seus respectivos posicionamentos
- Traçar as arestas das faces visíveis, que revelarão o objeto como enxergado de um determinado ponto de vista. Se desejar, é possível traçar também as linhas não visíveis naquele ponto de vista.

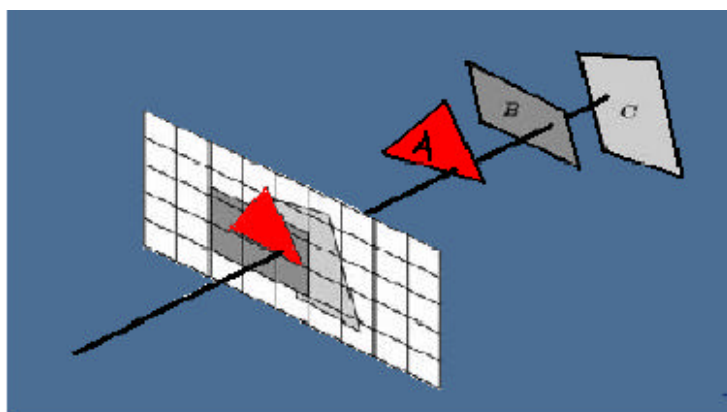
## Z-Buffer (ordenação por distância)

- Na representação eficiente de cenários complexos precisa-se de algoritmos de visualização para representar somente as partes visíveis do cenário.
- O z-buffer é o mais simples algoritmo de visualização.
- Geralmente implementado em nas placas de vídeo.
- Rotinas de OpenGL se baseiam neste algoritmo **Z**-



### Z-Buffer

- O Z-Buffer é uma matriz 2D que salva a componente de profundidade para cada pixel.
- Visibilidade é computada por pixel.
- O pixel recebe a cor do objeto mais próximo ao observador.



### Algoritmo z-buffer

1. Inicialize o plano com a cor de fundo
2. Inicialize o z-buffer com valores nulos:  
 $Z(x,y) = 0$ ; /\* plano de fundo \*/
3. Para cada polígono
4.     Para pixel projetado do polígono
5.          $pz$  = valor de  $z$  do polígono nas coords.  $(x,y)$  do pixel
6.         Se  $pz \geq Z(x,y)$
7.             Troque  $Z(x,y) = pz$
8.         Desenhe na tela o pixel  $(x,y)$  com a cor do polígono
9.         correspondente ao pixel projetado em  $(x,y)$