



## 2º Teste/1º Exame de Computação Gráfica

LEIC/LESIM

Prof. Mário Rui Gomes

6 de Julho de 2002

Nº \_\_\_\_\_ Nome: \_\_\_\_\_

Responda às questões seguintes justificando adequadamente todas as respostas. O Teste tem uma duração máxima, incluindo tolerância, de 1h30m. O Exame tem uma duração máxima, incluindo tolerância, de 2h.

Ao 2º teste pertencem todas as perguntas dos grupos V a IX. Ao 1º Exame pertencem todas as perguntas menos as dos grupos V a IX assinaladas como **[Teste]**.

### 1º Parte

#### I - Introdução e Conceitos Básicos

1. Na taxionomia das Aplicações Gráficas um dos critérios usados é o dos “Papéis Desempenhados pela Imagem”. Descreva, sucintamente, 2 desses tipos, dando um exemplo para cada um deles.

*Quanto aos papéis desempenhado pela imagem temos unicamente os seguintes 2 sub critérios:*

*1. Na maioria das aplicações de CG a cena composta por um ou mais objectos não é o objectivo final mas sim a criação de imagens ou sequência de imagens. A produção de publicações electrónicas, de pinturas ou de animações são exemplos desta situação. Por outro lado, para um cartógrafo o mais importante é o rigor da informação que é medida no Mundo Real e não a simples produção de mapas.*

*2. Já para a maioria dos engenheiros o objectivo é a criação da cena a qual representa um produto que irá ser produzido quer se trate de um edifício ou de um produto electromecânico.*

2. O que se entende por mapa de cor? Qual a redução de espaço, em percentagem, que se obtém através da utilização de um mapa com 256 cores com uma imagem com resolução de 360x270?

*Um mapa de cores é uma estrutura que declara as componentes de cor, usualmente RGB, das cores empregues pela imagem a que se encontra associada, atribuindo a cada cor um índice único. Cada cor declarada num mapa de cores ocupa 3 unidades de informação, ou seja, para 256 cores, o mapa de cores ocupará 768 unidades de informação. Se a informação de cada quadrícula for constituída pelo índice da respectiva cor, cada quadrícula ocupará apenas uma unidade de informação e não 3. Para a imagem das dimensões que estamos a considerar, passa a ser necessário armazenar 97.968 (768\*97.200) unidades de informação o que corresponde a uma percentagem de 33,6% do consumo original de memória, isto é, de 291.600 (3\*360\*270).*

3. O que se entende por atributos de uma primitiva gráfica? Desenhe o mesmo polígono 3 vezes, usando, de cada vez diferentes atributos.

*Um atributo é uma informação que é utilizada no controlo do modo como uma primitiva gráfica deve ser desenhada.*

*No primeiro exemplo da figura só é desenhado o contorno do polígono. No segundo exemplo é desenhado, com uma única cor, o seu interior o qual, no terceiro exemplo, é preenchido com um conjunto de segmentos de recta uniformemente espaçados. Neste último caso não é desenhada a fronteira da área poligonal. Outros atributos possíveis são a espessura da linha (ou da fronteira), a cor ou a textura a usar no preenchimento da área poligonal.*



## **II - Transformações/Projecções**

1. Que tipos de transformações têm lugar quando se transforma uma cena 3D no sistema de coordenadas do mundo para o espaço de coordenadas canónicas na situação em que o utilizador especificou uma projecção perspectiva? Indique a ordem pela qual essas transformações são realizadas.

*Primeiro, ocorre a transformação de visualização que corresponde a converter os objectos descritos em coordenadas do Mundo para coordenadas da câmara. É implementada por uma translação seguida de uma rotação. Depois, são realizadas duas transformações de escala para converter o volume de visualização genérico no volume canónico perspectivo. A primeira escala faz com que a inclinação dos planos laterais do frustum genérico tenham um declive unitário. A segunda escala passa o plano de recorte posterior de  $z=F$  para  $z=1$ . Poder-se-á ainda especificar a Transformação Perspectiva de modo a converter o volume canónico perspectivo num volume canónico ortogonal com o objectivo de simplificar as operações de visibilidade.*

### III - VRML

1. Considere o código seguinte (protótipo de uma vela com a chama animada):

```
01- PROTO PCandle [] {
02-   Group {
03-     children [
04-       DEF GFlame Transform {
05-         Translation 0 1.2 0
06-         children [
07-           Shape {
08-             geometry Sphere { radius .15 }
09-             appearance Appearance {
10-               material Material {
11-                 emissiveColor 1 1 0
12-                 transparency .5
13-               }
14-             }
15-           ]
16-         ]
17-       }
18-       DEF GStick Transform {
19-         children [
20-           Shape {
21-             geometry Cylinder {
22-               radius .1
23-               height 2
24-             }
25-             appearance Appearance {
26-               material Material {
27-                 diffuseColor 1 1 1
28-                 ambientIntensity .5
29-               }
30-             }
31-           ]
32-         ]
33-       }
34-     ]
35-   }
36-   DEF TS TimeSensor {
37-     loop TRUE
38-     cycleInterval .5
39-   }
40-   DEF PI PositionInterpolator {
41-     key [0 .5 1]
42-     keyValue [.2 1 .2, .4 1 .4, .2 1 .2]
43-   }
44-   ROUTE TS.fraction_changed TO PI.set_fraction
45-   ROUTE PI.value_changed TO GFlame.scale
46- }
```

a) Após ter instanciado o protótipo (só, sem outros nós adicionais) e desligado a “HeadLight”, verificou-se que apenas parte da cena estava visível. Qual e porquê?

*Apenas a chama está visível. Não existindo luzes na cena (tendo-se desligado a única que existe por omissão - headlight), apenas a componente emissiveColor dos materiais contribui para o cálculo da cor dos objectos geométricos.*

b) De forma a promover o realismo do protótipo, decidiu-se adicionar uma luz à vela. Indique qual nó VRML é o mais adequado para o efeito, justificando. Escreva o código adicional, indicando onde deverá ser inserido.

*O tipo de nó mais adequado é o nó PointLight, que simula uma fonte de luz com uma posição bem definida no espaço, irradiando em todas as direcções. O código seguinte deverá ser*

adicionado dentro do mesmo grupo onde está definida a geometria da chama – i.e. a seguir à linha 15:

```
DEF GFlameLight PointLight {
    on TRUE
    intensity 1
    ambientIntensity 1
    radius 10
}
```

2. Decididos a levar o realismo ainda mais longe, pretende-se agora fazer com que a intensidade da luz iluminando a cena também oscile ao ritmo da variação da geometria da chama (GFlame). Explique como implementar o efeito. Escreva o código adicional necessário e indique o seu ponto de inserção.

*Deverá ser adicionado um interpolador de intensidades (ScalarInterpolator) sincronizado com o interpolador de posição utilizado para fazer variar as dimensões da vela, através do mesmo TimeSensor. O código seguinte deverá ser introduzido a seguir à linha 45:*

```
DEF SI ScalarInterpolator {
    key [0 .5 1]
    keyValue [.1 .9 .1]
}
ROUTE TS.fraction_changed TO SI.set fraction
ROUTE SI.value_changed TO GFlameLight.intensity
```

## IV - Modelação

1. Enumere, descreva e justifique 4 propriedades que qualquer Esquema de Representação de Modelos Tridimensionais deve satisfazer.

*Em primeiro lugar, convém que as representações sejam **universais**, isto é, possam representar todos os objectos imagináveis e não apenas um número restrito. Como tal é impossível na prática, as representações deverão então poder representar o maior número possível de objectos.*

*Os objectos devem também ser fielmente representados. A fidelidade na representação significa que a representação deve ser tal que não existam ambiguidades de interpretação quanto ao objecto representado. Uma representação **sem ambiguidades** é uma representação completa se não forem omitidos quaisquer detalhes.*

*As representações devem ser únicas (**unicidade**), ou seja, cada tipo de representação não deve poder representar um objecto por mais do que uma forma. Isto permite simplificar operações tais como a comparação entre dois objectos. Se não houvesse uma representação única, poder-se-ia concluir que dois objectos eram diferentes embora se tratasse do mesmo objecto representado de duas maneiras.*

*A representação de objectos deve ser **precisa**, isto é, não deve conter aproximações pois disto depende o realismo da sua visualização (um cone com a aparência de pirâmide) e a correcção com que poderá ser manufacturado.*

2. Um dos Esquemas de Representação é designado por “Instanciação de Primitivas”. Descreva o Esquema de Representação, destacando o papel do uso de parâmetros. Indique um tipo de situação em que o Esquema é usado de modo adequado e dê um exemplo.

*Este tipo de representação tem por base a definição de objectos geométricos tridimensionais, as primitivas, que possuem atributos, os parâmetros, cujos valores são definidos pelo utilizador no momento da criação de uma nova instância.*

*Um exemplo simples de uma primitiva é o de um paralelepípedo com três parâmetros: comprimento, largura e altura. Ao variar os valores destes parâmetros, obter-se-ão paralelepípedos diferentes que são instâncias da mesma forma geométrica primitiva. O conceito de parâmetro não está confinado às dimensões. Assim, a primitiva pirâmide regular pode ter como parâmetro o número de faces laterais para além de parâmetros tipicamente geométricos como a sua altura e o raio da sua base. O conceito de parâmetro engloba ainda propriedades dos objectos reais a modelar tais como a definição dos materiais constituintes dos objectos, texturas e o acabamento das suas superfícies.*

*A instanciação de primitivas torna-se extremamente conveniente e simples quando as primitivas definem formas geométricas complexas, de definição difícil e morosa, e que não podem ser obtidas por operações lógicas tais como a adição ou subtracção de volumes, como são os casos de parafusos e rodas dentadas. Estes encontram-se normalizados e agrupados em séries também normalizadas. As séries definem características como a forma da cabeça dos parafusos ou dos dentes das rodas dentadas.*

3. Descreva a representação por “Enumeração da Ocupação do Espaço”. Descreva, justificando, uma vantagem deste tipo de representação.

*Na representação de sólidos por enumeração da ocupação do espaço cada sólido é representado por um conjunto de volumes idênticos que, conjuntamente, representam o volume ocupado pelo sólido a representar. A enumeração da ocupação do espaço decompõe o espaço segundo uma grelha tridimensional composta por volumes de forma e dimensões idênticos, os volumes elementares, também denominados por voxéis. A representação de um sólido consiste então em arbitrar a discretização pretendida, ou seja, determinar a dimensão dos voxéis, e, seguidamente, enumerar quais os voxéis ocupados pelo sólido a representar. Uma tal representação é única e não contém quaisquer ambiguidades.*

*O inconveniente mais grave desta representação consiste no número elevado de volumes elementares necessários, uma vez que este número cresce segundo uma lei cúbica quando se diminui a dimensão dos volumes elementares para aumentar a discretização do espaço, na tentativa de obter representações mais precisas dos sólidos. Com efeito, a redução para metade das dimensões dos voxels implica  $2^3 = 8$  mais voxels do que anteriormente.*

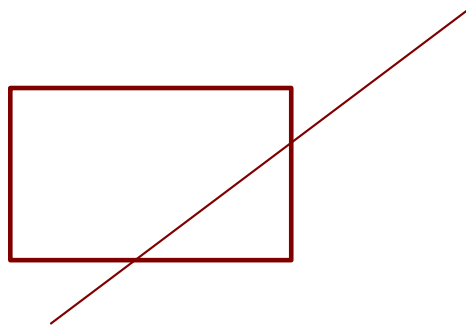
## 2º Parte (2º Teste)

### V - Pipeline de Visualização e Recorte

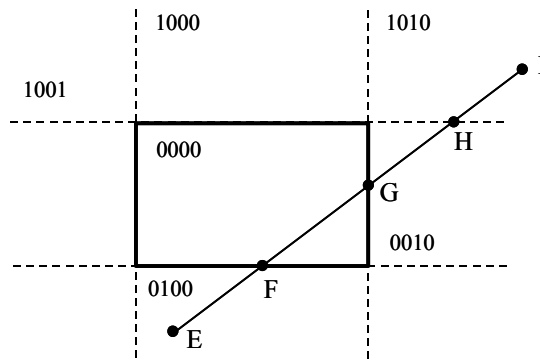
1. A *interface* com uma API gráfica baseada no modelo “Câmara Virtual Simples”, estudado nas aulas teóricas, exige do utilizador um conjunto de parâmetros que é responsável pela construção de uma determinada vista da cena. Que parâmetros são estes e qual a sua função?

*Os parâmetros são:*

- *VRP – Posição do observador ou da câmara;*
  - *VPN (normal ao plano de visualização) e VUV (up-vector que fixa a câmara em torno de vpn) definem a orientação da câmara;*
  - *As dimensões da janela de visualização (altura, largura ou os FOVs no caso de projecção perspectiva);*
  - *Distâncias  $D$ ,  $N$  e  $F$  (respectivamente a distância do observador à janela de vis.; ao plano anterior e ao plano posterior; todas elas medidas ao longo de vpn);*
  - *Tipo de projecção.*
2. Aplique o algoritmo de Cohen-Sutherland à figura. Inclua não só o valor dos outcodes mas também o critério adoptado na escolha de quais as intersecções que é necessário calcular.



*Considere-se a figura na qual é possível verificar que os outcodes são. E-0100, F-0000, G-0000, H-0010 e I-1010:*

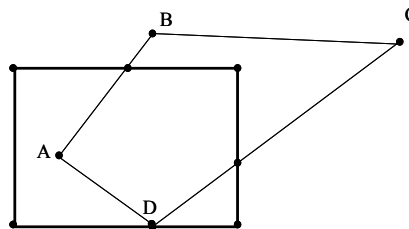


Como ambos os vértices estão fora do polígono de recorte usa-se o 1º vértice, E. Utilizando o primeiro bit a 1 do seu outcode efectua-se o recorte com a respectiva recta (inferior horizontal). Descarta-se EF.

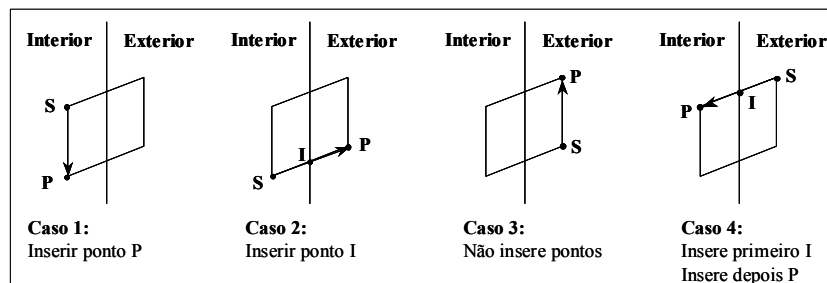
Como o outcode de F é 0000 vai utilizar-se o outcode do 2º vértice, I para escolher a nova recta a recortar. Como é o 1 bit então o recorte será efectuado com a recta superior horizontal do qual resulta o ponto H (outcode-0010) como resultado da intersecção. É descartado o segmento de recta HI.

Seguidamente utiliza-se o mesmo critério e recorta-se com a recta vertical direita, do que resulta G como ponto de intersecção, o qual tem outcode-0000 pelo que o segmento de recta FG é trivialmente aceite e a execução do algoritmo termina.

3. **[Teste]** Descreva o algoritmo de Sutherland – Hodgman e aplique-o ao polígono, da figura.



O algoritmo baseia-se na estratégia de dividir para conquistar: Assim é fornecido um primeiro conjunto de vértices e faz-se o recorte contra uma aresta infinita como indicado abaixo. Após o recorte será obtido um novo conjunto de vértices que são obtidos através das regras que se apresentam na figura.





*Considera-se, em primeiro lugar a recta de cima. Das intersecções obtém-se dois novos pontos B' (da intersecção com AB) e C' da intersecção com CD. Sendo o novo polígono AB'C'D.*

*No passo seguinte escolhe-se a recta da direita e obtém-se dois novos vértices, o vértice B'' (vértice superior direito do rectângulo de recorte) que resulta da intersecção com B'C' e o vértice C'' que resulta da intersecção com o C'D'. Resulta deste passo o polígono AB'B''C''D.*

*Por análise dos códigos dos vértices verifica-se que estão todos no interior do polígono de recorte pelo que o processo termina.*

## **VI – Rasterização**

1. Aplique o algoritmo de Bresenham ao segmento de recta  $P_0 = (5, 8)$   $P_1 = (9, 11)$ . Inclua, em cada passo a definição não só das coordenadas da quadrícula a desenhar mas também o valor da Variável de Decisão

*Os valores das diferenças em x e y são de  $dx = 4$  e  $dy = 3$  do que resulta, como 1º valor da variável de decisão  $d_0 = 2*(dy-dx)=2$  e  $incrE = 2*dy = 6$  e  $incrNE = 2*(dy-dx) = -2$ .*

*A primeira quadrícula a ser escrita é a inicial (5, 8).*

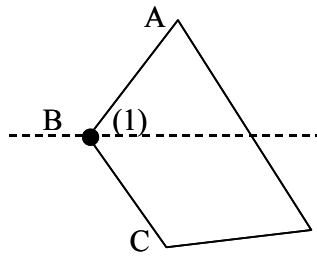
*Como a variável de decisão inicial é 2 ( $>0$ ) tem-se uma evolução em NE, escreve-se a quadrícula (6, 9) e o novo valor da variável de decisão será  $d_1 = 2-incrNE = 0$ .*

*Os valores que se obtém nos passo seguintes são:*

- $d_1 = 0 \Rightarrow E \Rightarrow Write (7, 9)$
- $d_2 = 6 \Rightarrow NE \Rightarrow Write (8, 10)$
- $d_3 = 4 \Rightarrow NE \Rightarrow Write (9, 11)$

2. No preenchimento de polígonos côncavos com fronteira composta por segmentos de recta, recorrendo à coerência espacial em linha de varrimento é fundamental que não existam erros na identificação de intersecções. Como se contabilizam as intersecções quando a linha de varrimento coincide com um vértice? Dê um exemplo.

*Uma linha de varrimento ao passar por um vértice intermédio, contabiliza intersecção na aresta intersectada em  $Y_{min}$  e não na aresta intersectada em  $Y_{max}$ .*



1 intersecção com [AB]  
0 intersecções com [BC]

## VII – Remoção

- Os algoritmos de Remoção de Elementos Ocultos pode ser subdivididos em duas categorias. Diga quais são, como funcionam e caracterize cada uma delas quanto ao *aliasing* e à complexidade algorítmica.

*As categorias são:*

- Precisão Imagem que segue uma estratégia de determinar qual o objecto mais próximo visível através de uma quadrícula e desenhá-la com a cor desse objecto. Para  $n$  objectos e  $p$  quadrículas, complexidade algorítmica  $(n \times p)$ . Estes algoritmos introduzem *aliasing* e para  $n$  objectos e  $p$  quadrículas têm, normalmente uma complexidade algorítmica de  $n * p$ .*
  - Precisão Objecto que segue uma estratégia de determinar as partes visíveis de um objecto e desenhar só essas partes com a cor do objecto. Estes algoritmos não introduzem *aliasing* e para  $n$  objectos e  $p$  quadrículas têm, normalmente uma complexidade algorítmica  $n^2$ .*
- É possível comparar famílias de algoritmos através do modo como é efectuada a ordenação dos objectos da cena. Indique, justificando, como é efectuada tal ordenação nos algoritmos de Ordenação em Profundidade, Linha de Varrimento e Z-Buffer.

**Ordenação em Profundidade**, tal como o nome indica é efectuada uma ordenação em Z e, para resolver ambiguidades é analisada as relações entre as coordenadas X e Y de todos os polígonos que ocupam intervalos segundo  $z'$  se sobrepõem. Por vezes é necessário efectuar a subdivisão dos polígonos.

**Linha de Varrimento**, tal como o nome indica é efectuada uma ordenação por linha, isto é em Y, seguida em X e quando co-existem vários polígonos é necessário calcular o respectivo valor de Z para saber qual o polígono que será desenhado.

No **Z-Buffer** não existe qualquer tipo de ordenação. Para cada quadrícula e cada polígono vai sendo armazenado o valor da cor do polígono mais próximo do observador assim como a respectiva cor da quadrícula.

3. **[Teste]** Descreva, sucintamente, os algoritmos de Ordenação em Profundidade e o Algoritmo do Pintor.

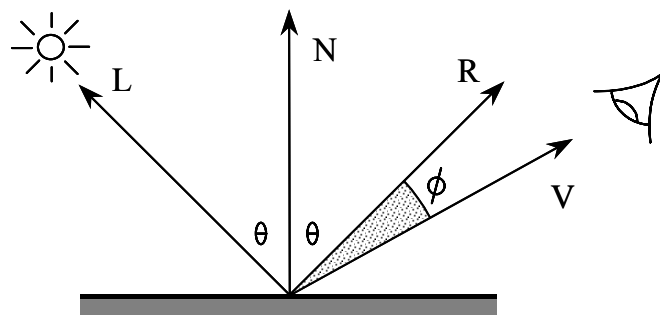
O objectivo é desenhar os polígonos na ordem decrescente da distância ao observador. Para tal, em primeiro lugar todos os polígonos constituintes da cena por ordem crescente da sua menor coordenada  $z$ . Seguidamente resolvem-se situações ambíguas, que consistem nos casos em que os intervalos de ocupação dos polígonos segundo  $zz'$  se sobrepõem. Por vezes, a resolução dessas ambiguidades passa pela fragmentação dos polígonos. Finalmente discretizam-se os polígonos (originais e resultantes da fragmentação) por ordem crescente da sua menor coordenada.

O Algoritmo do Pintor é uma versão simplificada do algoritmo anterior que se aplica à situação em que todos os polígonos pertencem a planos com  $Z$  constante (2.5D), pelo que não ocorrem situações de ambiguidade.

## VIII – Cor, Sombreamento e Formatos (8)

1. Escreva a fórmula do Modelo de Reflexão de Phong e descreva o significado de cada um dos componentes. Deve usar uma figura para ilustrar o significado de alguns dos componentes.

Considere-se a seguinte imagem:



A Fórmula de Iluminação de Phong é:

$$I = I_a \cdot K_a + I_i [ K_d (L \cdot N) + K_s (R \cdot V)^n ]$$

Em que:

- $I$  é Intensidade da Energia Luminosa Reflectida pelo objecto.

- *Ia é a Intensidade da Energia Luminosa Ambiente.*
  - *Ii é a Intensidade da Energia Luminosa Incidente.*
  - *Ka é constante para um dado objecto (Coeficiente de Reflexão de Luz Ambiental)•*
  - *Kd é a constante de reflectividade difusa (0.0,1.0) e traduz o modo como o material reage ao comprimento de onda da luz incidente.*
  - *Ks é o coeficiente de reflexão especular, de valor constante e dependente das propriedades ópticas do material.*
  - *n (“shininess”) traduz o grau de brilho.*
2. Apresente 2 exemplos de problemas na imagens de objectos obtidas através da aplicação de métodos de sombreamento interpolados. Se entender por recorrer a desenhos para suportar a explicação do problema.

*Podem ser escolhidos quaisquer dois dos seguintes:*

- *Aspecto marcadamente poligonal ao longo dos lados que constituem a silhueta de superfícies curvas, aproximadas por malhas de polígonos.*
  - *A interpolação de intensidades, constante ao longo das linhas de varrimento, não está de acordo com a distorção provocada pela transformação de perspectiva.*
  - *O resultado do sombreamento depende da orientação dos polígonos. Perturbações quando a orientação de um objecto varia lentamente ao longo da sequência de imagens de um trecho de animação.*
  - *Nem sempre as normais calculadas nos vértices da malha de polígonos constituem uma boa aproximação à normal da superfície a sombrear (refinar a aproximação poligonal).*
3. Que outra designação tem o formato BMP? Qual a sua origem? Descreva 3 das suas características principais.

*O formato DIB (Device Independent Bitmap), também designado por formato BMP, é um formato proprietário da Microsoft e é suportado por todas as variantes do sistema operativo Windows.*

*Na origem da designação DIB está o facto de que este formato descreve as cores de uma forma que é independente do processo empregue por cada placa gráfica e respectivo monitor para apresentar a cor das quadrículas das imagens. No formato DIB, as imagens são descritas por varrimento ascendente das linhas (de baixo para cima), sendo as quadrículas de cada linha*

varridos da esquerda para a direita. Do varrimento de cada linha deve resultar uma sequência de bytes cujo número deve ser múltiplo de 4.

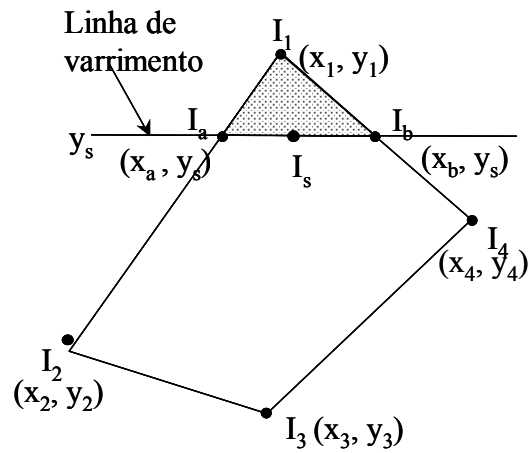
O formato DIB permite descrever imagens a cores com 1, 4, 8 ou 24 bits por quadrícula, representando assim imagens com 2, 16, 256 ou  $2^{24}$  cores, respectivamente, empregando um mapa de cores em todos os casos excepto no último ( $2^{24}$  cores). O formato DIB permite ainda a compressão opcional do conteúdo de imagens com 16 ou 256 cores pelo algoritmo RLE (Run Length Encoding) adaptado ao número de bits por quadrícula (4 ou 8).

Resumindo as suas características:

- Cores: 2, 16, 256 e 16.777.216
- Modelo de cor: RGB
- Compressão: RLE 4 e RLE 8
- Mapa de cores: só com 2, 16 e 256 cores
- Entrelaçamento: não suporta
- Varrimento vertical: ascendente
- Transparência: não suporta

4. **[Teste]** No trabalho de laboratório aplicou o sombreado de Gouraud a qualquer polígono convexo com fronteira formada por segmentos de recta. Considere a simplificação ao modelo em que este é unicamente composto por triângulos. Descreva sucintamente os cálculos efectuados. Justifique a existência de brilho, por exemplo, no corpo do bule de chá (*teapot*).

*Na leitura do ficheiro obtinham-se não só as coordenadas dos vértices de cada polígono mas também a respectiva cor a qual, supostamente, deveria ter sido obtida através da aplicação do Modelo de Iluminação de Phong à normal ao vértice (obtida através da interpolação das normais a cada uma das facetas que partilham esse mesmo vértice).*



*Na realidade o modelo de sombreado de Gouraud não conduz à obtenção do que se designa por brilho, no entanto, o facto dos triângulos terem uma dimensão muito pequena conduziu a que o factor preponderante fosse a interpolação de normais, como já descrito. Assim estamos na situação em que quase não está a ser usada uma técnica incremental de sombreado.*

## IX – Ray-Tracing e Radiosidade

1. Diga o nome de um dos investigadores que esteve na origem do algoritmo de Ray-Tracing, assim como qual foi a sua contribuição científica. Descreva, sucintamente e recorrendo a uma figura o Algoritmo Fundamental de Ray-Tracing.

*Na base de todos os algoritmos de ray tracing está o trabalho de Appel desenvolvido em 1968 com o intuito de determinar sombras, nomeadamente se um ponto visível de uma cena estaria ou não no interior de uma zona de sombra. Entre 1968 e 1971, Goldstein e Nagel desenvolveram algoritmos para o cálculo de trajetórias balísticas e de partículas nucleares que, só depois, conjuntamente com os algoritmos desenvolvidos por Appel, foram aplicados à geração de imagens. Em 1979, Kay introduziu as modificações necessárias para o cálculo mais correcto da iluminação. especular e da refração da luz, tendo este último aspecto sido completado por Whitted em 1980.*

*Na sua origem, os algoritmos de ray tracing são algoritmos para determinar a visibilidade dos objectos de uma cena, operando ao nível de precisão da imagem. O algoritmo fundamental de ray tracing considera raios com origem no centro de projecção em que cada um dos raios passa pelo ponto correspondente a uma quadrícula da imagem sobre o plano de projecção da cena. Se um raio não intersectar nenhum objecto da cena, é atribuída à quadrícula da imagem por onde o raio passa a cor de fundo da cena. Caso contrário, determinam-se as intersecções do raio com os objectos da cena e a quadrícula correspondente ao raio é atribuída a cor do ponto de intersecção mais próximo da origem do raio.*

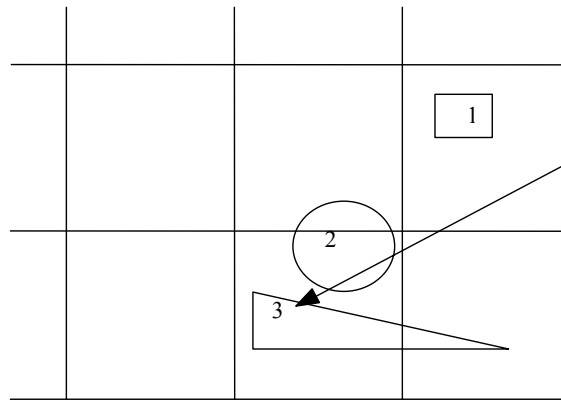
2. Descreve sucintamente o Método da Radiosidade? O que se entende por Absorsividade, Reflectividade, Transmissividade?

*Na base do método da radiosidade encontra-se o balanço de energia entre a radiação que atinge uma superfície e a radiação que deixa a mesma superfície. Estas duas quantidades de energia devem ser iguais numa situação de equilíbrio.*

*A energia radiante total que atinge uma dada superfície, a irradiação total da superfície, pode ser por esta absorvida, reflectida ou transmitida em proporções que dependem das propriedades físicas da superfície e do respectivo corpo. A quantificação destas proporções está ligada a três propriedades da superfície que são*

- *Absorsividade ( $a$ ) – fracção da irradiação total que é absorvida;*
- *Reflectividade ( $r$ ) – fracção da irradiação total que é reflectida;*
- *Transmissivida de ( $t$ ) – fracção da irradiação total que é transmitida através do corpo da superfície.*

3. **[Teste]** Uma das técnicas de optimização do Algoritmo de Ray-Tracing recorre à Partição no Espaço. Descreva sucintamente a técnica recorrendo ao exemplo na figura na qual se pode observar um raio e três objectos.



*Na técnica da Partição do Espaço este é dividido em partições iguais que desempenham o papel de volumes envolventes. Um objecto pode localizar-se em mais do que uma partição. Dado um raio, determina-se a sua intersecção com as partições seguindo uma ordem de teste por distância crescente das partições à origem do raio.*

*Uma partição intersectada pelo raio é liminarmente eliminada se não contiver quaisquer objectos. Caso contrário, calculam-se então as intersecções do raio com os objectos contidos na partição para, no final, reter a intersecção correspondente à menor distância à origem do raio, aplicando a cada partição uma lógica em tudo semelhante à aplicada a toda a cena quando se não considera a sua partição. No caso em que um objecto se encontra contido em mais do que*

*uma partição do espaço (caso dos objectos 2 e 3) é necessário verificar se a intersecção determinada ocorre na partição corrente e que deu origem ao desencadear dos cálculos de intersecção do raio com os objectos contidos na partição. A razão desta verificação adicional é que a intersecção com o objecto poderá ocorrer numa outra partição que não a partição para a qual se realizaram os cálculos de intersecção.*

*Aplicando o algoritmo ao exemplo da figura a primeira partição contém o objecto 1. Como a intersecção não existe o algoritmo continua a ser executado.*

*Seguidamente escolhe-se a partição que contém parcialmente o objecto 3. Nesse caso o ponto de intersecção ocorre numa outra partição pelo que o objecto '2 "marcado" e algoritmo continua a ser executado. Na partição seguinte é calculada a intersecção só com o objecto 2, uma vez que o objecto 3 já foi "marcado" como tendo sido calculada a sua intersecção com o raio. Como a intersecção com esse objecto está contida na partição que está a ser tratada e as suas coordenadas estão mais próxima da origem do raio a execução do algoritmo termina.*



### Cotação do 2º Teste

	V	VI	VII	VIII	IX	Total
1.	1	2	1	1,5	1	
2.	2	1	1,5	1	1,5	
3.	1		1,5	1	1,5	
4.				1,5		
<b>Total</b>	<b>4</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>4</b>	<b>20 Valores</b>

### Cotação do 1º Exame

	I	II	III	IV	V	VI	VII	VIII	IX	Total
1.	0,5	1	1,5	1	0,5	1,5	1	1	0,5	
2.	1		1	1	1,5	0,5	1	1	1,5	
3.	1			1				1		
<b>Total</b>	<b>2,5</b>	<b>1</b>	<b>2,5</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>20 Valores</b>