

Sistemas Gráficos e Interação

Época Normal

2013-01-31

N.º _____ Nome _____

Duração da prova: 75 minutos

Cotação de cada pergunta: assinalada com parêntesis rectos

Perguntas de escolha múltipla: cada resposta incorrecta desconta 1/3 do valor da pergunta

Parte Teórica

30%

a. [2.5] Os processadores/aceleradores gráficos

- i. Recorrem a técnicas de paralelismo para atingir elevados desempenhos
- ii. Aliviam o processador central de diversas tarefas tais como transformações, recorte, projecção e mapeamento de texturas
- iii. Possuem normalmente memória dedicada
- iv. Todas as anteriores

b. [2.5] A realização de uma translação de um ponto P para um ponto P' distinto do anterior com base numa simples multiplicação de matrizes

$$P' = T \times P$$

- i. Não é possível; seria necessária uma adição de matrizes que reflectisse o deslocamento de P para P'
- ii. É possível desde que a matriz T seja a matriz identidade
- iii. É possível desde que se usem coordenadas homogéneas
- iv. Nenhuma das anteriores

c. [2.5] A qual das seguintes sequências de transformações corresponde uma matriz de transformação composta igual à matriz identidade?

- i. `glTranslated(1.0, 2.0, 3.0); glRotated(0.0, 0.0, 0.0, 1.0);`
- ii. `glRotated(45.0, 0.0, 0.0, 1.0); glScaled(1.0, 1.0, 1.0);`
- iii. `glScaled(2.0, 4.0, 8.0); glTranslated(0.0, 0.0, 0.0);`
- iv. Nenhuma das anteriores

d. [2.5] Admita que pretende aplicar a um objecto a seguinte sequência de transformações: uma translação $T1$, seguida de uma translação $T2$. Qual das seguintes afirmações é correcta?

- i. A matriz $T1 \times T2$ traduz a composição de transformações pretendida; a matriz $T2 \times T1$ não traduz, pois a multiplicação de matrizes não goza da propriedade comutativa
- ii. A matriz $T2 \times T1$ traduz a composição de transformações pretendida; a matriz $T1 \times T2$ não traduz, pois a multiplicação de matrizes não goza da propriedade comutativa
- iii. Ambas as matrizes traduzem a composição pretendida, pois a composição de translações goza da propriedade comutativa
- iv. Nenhuma das anteriores

- e. **[2.5]** Na representação de sólidos por fronteira (B-Rep – *Boundary Representation*)
- O modelo é representado com base na divisão do espaço em cubos cujos lados são potências de base 2
 - O modelo é representado por uma árvore em que os nós internos designam operações de conjuntos ou transformações lineares afim e as folhas denotam objectos primitivos
 - O modelo é representado através da superfície que o delimita
 - Há ambiguidade, pois a uma mesma representação podem corresponder vários modelos
- f. **[2.5]** O modelo de iluminação do OpenGL
- Constitui um exemplo de um modelo de iluminação global
 - Dispensa a especificação de normais
 - Contempla apenas os caminhos para a luz do tipo fonte luminosa → superfície → observador
 - Nenhuma das anteriores
- g. **[2.5]** Uma forma de determinar o vector normal a um polígono planar consiste em
- Calcular o produto escalar dos vectores definidos por duas arestas do polígono e dividir o vector resultante pelo seu comprimento
 - Projectar o polígono nos planos OYZ, OZX e OXY e calcular as áreas dos polígonos resultantes; as componentes da normal serão proporcionais a estes valores
 - Dividir o polígono em dois ou mais triângulos
 - Nenhuma das anteriores
- h. **[2.5]** A função de mapeamento de texturas que a seguir se discrimina baseia-se numa parametrização

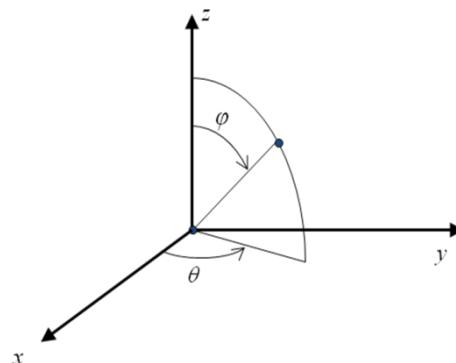
$$x(\varphi, \theta) = \sin \varphi \cos \theta$$

$$y(\varphi, \theta) = \sin \varphi \sin \theta$$

$$z(\varphi, \theta) = \cos \varphi$$

$$\varphi = \pi \cdot t$$

$$\theta = 2\pi \cdot s$$



- Esférica
- Cilíndrica
- Cúbica
- Nenhuma das anteriores

Sistemas Gráficos e Interacção

Época Normal

2013-01-31

N.º _____ Nome _____

Parte Teórico-Prática

40%

Perguntas de escolha múltipla: cada resposta incorrecta desconta 1/3 do valor da pergunta
Nota: Em todas as perguntas, a menos que algo seja dito em contrário, assuma a posição da câmara por omissão.

- a. Admita que se pretende controlar a câmara livremente na cena, podendo o utilizador deslocá-la para a frente e para trás, rodá-la para a esquerda e para a direita, bem como rodá-la para cima e para baixo.

- i. **[2.0]** Qual a estrutura de dados que definiria para o modelo desta câmara?

- ii. **[3.0]** Quais os cálculos a efectuar para posicionar a câmara na cena com a função `gluLookAt()`?

`float eyex =` _____

`float eyey =` _____

`float eyez =` _____

`float centerx =` _____

`float centery =` _____

`float centerz =` _____

`gluLookAt(eyex, eyey, eyez, centerx, centery, centerz, 0.0, 1.0, 0.0);`

Sistemas Gráficos e Interacção

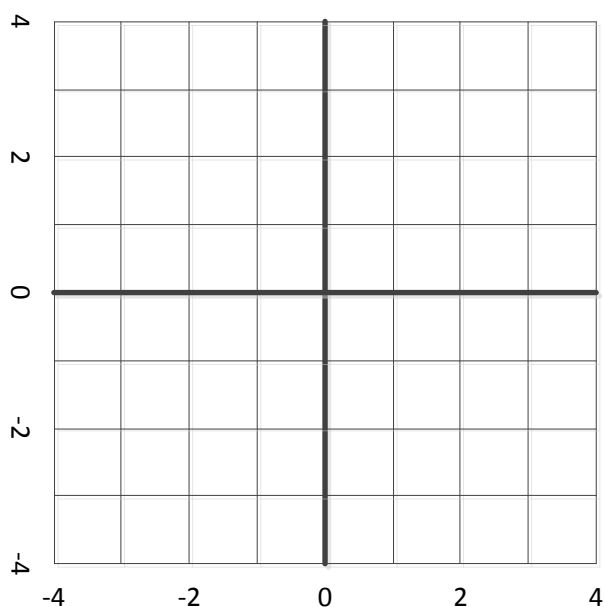
Época Normal

2013-01-31

N.º _____ Nome _____

b. **[3.0]** Esboce o desenho que resulta do seguinte extracto de código.

```
glBegin(GL_TRIANGLE_FAN);  
    glVertex2d(-1.0, -1.0);  
    glVertex2d(2.0, 1.0);  
    glVertex2d(0.0, 1.0);  
    glVertex2d(-1.0, 1.0);  
    glVertex2d(-2.0, 1.0);  
glEnd();
```



- c. **[4.0]** Admita que pretende iluminar uma cena com uma luz ambiente a 30% e um foco móvel de cor azul a 100%. O foco é representado pela variável global `spot`. No extracto de código que se segue, assinale com um círculo as linhas de código necessárias para configurar a situação pretendida.

```
void init()
{
    GLfloat light_gray[] = { 0.3, 0.3, 0.3, 1.0 };
    GLfloat light_blue[] = { 0.0, 0.0, 1.0, 1.0 };
    GLfloat light_position[] = {spot.x, spot.y, spot.z, 1.0};
    GLfloat light_direction[] = {spot.dirx, spot.diry, spot.dirz};
    GLfloat light_cutoff[] = { 90.0 };

    ...
    glLightfv(GL_LIGHT0, GL_AMBIENT, light_gray);
    glLightfv(GL_LIGHT0, GL_DIFFUSE, light_blue);
    glLightfv(GL_LIGHT0, GL_POSITION, light_position);
    glLightfv(GL_LIGHT0, GL_SPOTDIRECTION, light_direction);
    glLightfv(GL_LIGHT0, GL_CUTOFF, light_cutoff);
    glEnable(GL_LIGHT0);

    ...
    glLightfv(GL_LIGHT1, GL_AMBIENT, light_gray);
    glLightfv(GL_LIGHT1, GL_DIFFUSE, light_blue);
    glLightfv(GL_LIGHT1, GL_POSITION, light_position);
    glLightfv(GL_LIGHT1, GL_SPOTDIRECTION, light_direction);
    glLightfv(GL_LIGHT1, GL_CUTOFF, light_cutoff);
    glEnable(GL_LIGHT1);

    ...
    glLightModel(GL_LIGHT_MODEL_AMBIENT, light_gray);
}

void display()
{
    GLfloat light_gray[] = { 0.3, 0.3, 0.3, 1.0 };
    GLfloat light_blue[] = { 0.0, 0.0, 1.0, 1.0 };
    GLfloat light_position[] = {spot.x, spot.y, spot.z, 1.0};
    GLfloat light_direction[] = {spot.dirx, spot.diry, spot.dirz};
    GLfloat light_cutoff[] = { 90.0 };

    ...
    glLightfv(GL_LIGHT1, GL_AMBIENT, light_gray);
    glLightfv(GL_LIGHT1, GL_DIFFUSE, light_blue);
    glLightfv(GL_LIGHT1, GL_POSITION, light_position);
    glLightfv(GL_LIGHT1, GL_SPOTDIRECTION, light_direction);
    glLightfv(GL_LIGHT1, GL_CUTOFF, light_cutoff);
    glEnable(GL_LIGHT1);

    ...
}
```

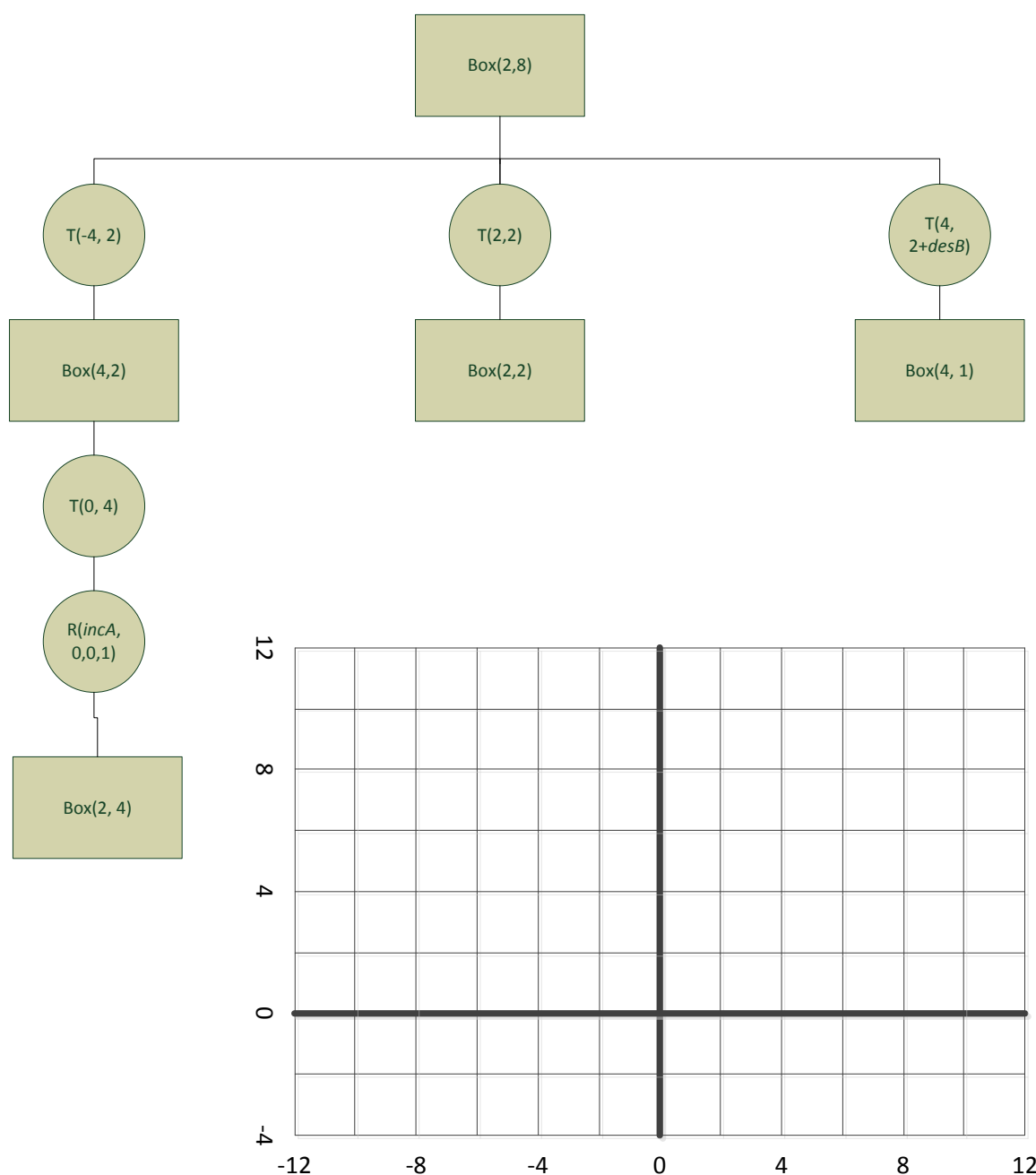
Sistemas Gráficos e Interação

Época Normal

2013-01-31

N.º _____ Nome _____

- d. **[5.0]** Assuma que uma determinada cena é desenhada de acordo com a árvore de cena ilustrada na figura. A função $\text{Box}(A, C)$ desenha um retângulo de altura A e comprimento C , com a origem no canto inferior esquerdo. Reproduza a cena no plano XY, considerando as alturas no semieixo positivo dos YY, e assumindo que $\text{incA} = 90.0$ e $\text{desB} = 1.0$.



Sistemas Gráficos e Interação

Época Normal

2013-01-31

N.º _____ Nome _____

- e. **[3.0]** Considere uma cena composta por um único objecto. Pretende-se realizar uma animação automática que consiste em deslocar o objecto na direcção horizontal, entre -5.0 e +5.0. Assuma a existência da variável `modelo.desloc` e escreva o código que permite efectuar a referida animação.

```
void Display()
{
    glClear(GL_COLOR_BUFFER_BIT);

    glLoadIdentity();

    _____

    _____

    DesenhaObjecto();

    glFlush();
}
```

```
void Timer(int value)
{
    glutTimerFunc(100, Timer, 0);

    _____

    _____

    _____

    _____

    _____

    _____

    // redesenhar o ecrã

    glutPostRedisplay();
}
```