Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina: Computação Gráfica
AP1 - 1° semestre de 2009.

1) Defina uma triangulação no plano (bidimensional) especificando as condições que devem ser satisfeitas. (2 pontos)

Uma triangulação de uma região do plano é uma coleção $T = \{T_i\}$ de triângulos de modo que dados dois triângulos distintos T_i e T_j ($T_i \cap T_j \neq Ø$), na triangulação, tem-se:

1. $T_i \cap T_j$ = vértice comum, ou

2. $T_i \cap T_j$ = é uma aresta comum

A existência de uma triangulação numa região permite a definição de atributos da região de forma localizada em cada triângulo. Cada triângulo possui um sistema natural de coordenadas locais, definido pelas coordenadas baricêntricas. Esse sistema permite, por exemplo, que atributos definidos nos vértices da triangulação possam ser interpolados linearmente para reconstruir os atributos em toda a região.

2) Considere um conjunto de quatro (4) pontos de controle P = {(1.0,1.0),(2.0,4.0),(4.0,4.0),(5.0,2.0)}. Determine as coordenadas (x,y) de um ponto de uma curva de Bézier para o valor do parâmetro u=0.5.Considere a seguinte expressão abaixo onde Px_i e Py_i são as coordenadas x e y do ponto de controle P_i e B_i as bases de Bézier: (2 pontos)

$$x(u) = \sum_{i=0}^{n} Px_i B_i(u)$$

$$y(u) = \sum_{i=0}^{n} Py_i B_i(u)$$

$$x(u) = \sum_{i=0}^{3} \binom{n}{i} u^i (1-u)^{n-i} x(i)$$

$$y(u) = \sum_{i=0}^{3} \binom{n}{i} u^i (1-u)^{n-i} y(i)$$

$$x(u) = \binom{3}{0} u^0 (1-u)^3 x(0) + \binom{3}{1} u^1 (1-u)^2 x(1) + \binom{3}{2} u^2 (1-u)^1 x(2) + \binom{3}{3} u^3 (1-u)^0 x(3)$$

$$y(u) = \binom{3}{0} u^0 (1-u)^3 y(0) + \binom{3}{1} u^1 (1-u)^2 y(1) + \binom{3}{2} u^2 (1-u)^1 y(2) + \binom{3}{3} u^3 (1-u)^0 y(3)$$

$$x(u) = (1-u)^3 u^0 x(0) + 3u(1-u)^2 x(1) + 3u^2(1-u)x(2) + u^3 x(3)$$

$$y(u) = (1-u)^3 u^0 y(0) + 3u(1-u)^2 y(1) + 3u^2(1-u)y(2) + u^3 y(3)$$

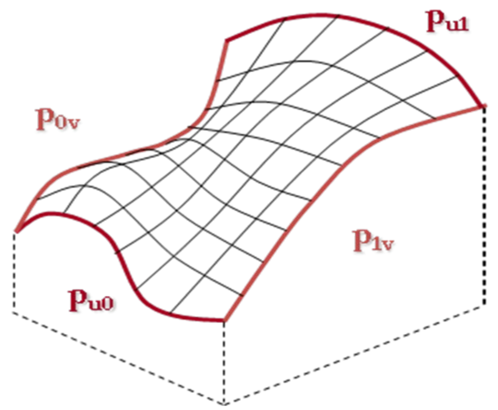
$$x(u) = (1-u)^3 u^0 1.0 + 3u(1-u)^2 2.0 + 3u^2(1-u)4.0 + u^3 5.0$$

$$y(u) = (1-u)^3 u^0 1.0 + 3u(1-u)^2 4.0 + 3u^2(1-u)4.0 + u^3 2.0$$

$$x(0.5) = (0.5)^3 1.0 + (0.5)^3 6.0 + (0.5)^3 12.0 + (0.5)^3 5.0 = 3.0$$

$$y(0.5) = (0.5)^3 1.0 + (0.5)^3 12.0 + (0.5)^3 12.0 + (0.5)^3 2.0 = 3.375$$

3) Considere um retalho representado por 4 curvas pu0,pu1,pv0,pv1. Determine a expressão para reconstrução do retalho C(u,v) utilizando o esquema de parametrização de Coons. (2 pontos)



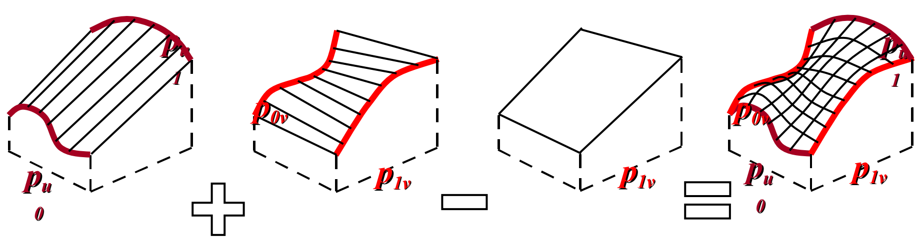
O método denominado Parametrização de Coons. Consiste em combinar diversas interpolações lineares das curvas de bordo segundo o seguinte esquema:

1. *Cálculo do Lofting vertical – interpolamos linearmente as curvas* p_{u0} *e* p_{u1}. $(1\text{-}v)p_{u0}(u)+vp_{u1}(u)$

2. *Cálculo do Lofting horizontal – interpolamos linearmente as curvas* p_{0v} *e* p_{1v}. $(1\text{-}u)p_{0v}(v)+up_{1v}(v)$

3. *Soma dos dois loftings – somamos as operações de lofting horizontal e vertical obtendo a parametrização* $C'(u,v)=(1\text{-}v)p_{u0}(u)+vp_{u1}(u)+(1\text{-}u)p_{0v}(v)+up_{1v}(v)$

4. *Subtração da parametrização C'(u,v) da interpolação bilinear B(u,v) dos vértices* p_{00}, p_{10}, p_{11} *e* p_{01}. *Como resultado, obtemos a parametrização de coons dada por:* $C(u,v) = C'(u,v)\text{-}B(u,v)$



4) O que é um *game engine*. Quais são as facilidades que tal ferramenta oferece ao desenvolvedor? (2 pontos)

Uma game engine é um sistema de software voltado para criação de jogos eletrônicos. Uma game engine ou motor de jogos é composta por várias componentes tais como, motores de rendering, motor de fisica e detecção de colisões, módulo para áudio, interação, scripting, animação, componente de inteligência artificial, comunicação em rede, gerenciamento de memória, threading, grafo de cena e outros.

Algumas game engines fornecem um conjunto de ferramentas visuais de desenvolvimento integradas, além de componentes de software reutilizáveis, o que que permite um desenvolvimento simples e rápido voltado ao conteúdo do jogo.

5) Como são especificadas as primitivas na OpenGL. Exemplifique com um caso. (2 pontos)

As primitivas da OpenGL são especificadas através do comando

begin (tipo_da_primitiva)
<seqüência de comandos para especificação de vértices ou atributos de vértices>
end;

Um exemplo de primitiva é um quadrilátero conforme especificado abaixo:

```
glBegin(GL_QUADS);                    /* Inicia o desenho da primitiva poligono */
        glVertex2f(-0.5, -0.5);              /* Primeiro vertice */
        glVertex2f(-0.5, 0.5);
        glVertex2f(0.5, 0.5);
        glVertex2f(0.5, -0.5);               /* Ultimo vertice */
  glEnd();                                  /* Encerra desenho da primitiva */
```