

Sexta Lista de Exercícios – Computação Gráfica

Curso de Ciência da Computação – Primeiro Semestre de 2010

Questões – Curvas e Superfícies

1. Considere um sistema de *coordenadas cartesiano tridimensional* associado a um SRU e implemente uma função para determinar pontos arbitrários de uma curva representada por $P(t) = (x(t), y(t), z(t))$, onde $0 \leq t \leq 1$. Para isso, considere a abordagem das curvas de *Bézier*. O usuário deverá entrar com o número de pontos de controle e as coordenadas de cada um destes pontos, respectivamente. Considere na implementação os casos onde os pontos de controle são igualmente espaçados e depois o caso onde os pontos dados pelo usuário não possuem um espaçamento uniforme. Tente descrever curvas com regiões de curvatura elevada. A saída da função deve ser apenas o valor das coordenadas da curva no ponto desejado. Avalie sua função para diversas entradas.
2. Da mesma forma que no exercício anterior, implemente uma função para determinar pontos arbitrários de uma curva representada por $P(t)=(x(t),y(t),z(t))$. Contudo, considere a abordagem das curvas *B-Splines*. O usuário deve entrar com o número de pontos de controle e os pontos de controle, respectivamente, associados a cada nó (t_0, t_1, \dots) da curva. O parâmetro k , que controla a ordem de continuidade da curva, também deve ser dado pelo usuário. Como antes, avalie sua função considerando pontos equidistantes e pontos de controle que não sejam uniformemente espaçados. A saída da função deve ser apenas o valor das coordenadas da curva no ponto desejado.
3. (Opcional) Implemente uma função que determine um ponto do espaço que pertence a uma superfície (imagine uma!). Considere a abordagem das superfícies de *Bézier*.
4. (Opcional) Repita o exercício anterior considerando a abordagem das superfícies *B-Spline*.
5. Considere um SRU e um sistema de coordenadas cartesiano tridimensional. Construa dois objetos retangulares neste espaço. Sendo dados pelo usuário um centro de projeção e um plano de projeção, implemente uma função que projete os pontos dos objetos na janela de projeção. Finalmente, implemente o algoritmo *z-buffer*, para eliminação de superfícies ocultas, e mostre o resultado na tela do computador utilizando o ambiente X-Window.