

MC930A – Computação Gráfica - 2017-S1 - Jorge Stolfi

Trabalho de laboratório 05 - 2017-05-12

Oitenta Tentáculos - Tentativa II

| | | |
|------|----|------|
| Nome | RA | Nota |
|------|----|------|

Objetivos. Treinar a construção de splines de Bézier suaves.

Enunciado. Sua tarefa nesta aula prática é continuar o projeto da aula anterior, a modelagem de *um monstro com um monte de tentáculos* em POV-Ray. Especificamente, o objetivo é construir cada tentáculo pela concatenação suave de vários segmentos como o construído na aula anterior.

Teoria: Lembramos que cada segmento tem como eixo uma curva de Bézier $p(t)$, definida por um intervalo de tempo $[t_0 - t_1]$ e quatro *pontos de controle* pA, pB, pC, pD , e tem raio que varia segundo um polinômio de Bézier $r(t)$ definido pelo intervalo $[t_0 - t_1]$ e quatro *valores de controle* rA, rB, rC, rD .

Lembramos também a condição para a emenda suave de duas curvas de Bézier $p(t)$, $q(t)$ definidas para intervalos de tempo iguais $[t_0 - t_1]$ e $[t_1 - t_2]$, respectivamente. Sejam pA, pB, pC, pD e qA, qB, qC, qD os pontos de controle das duas curvas. A emenda é suave se $pD = qA = (pC + qB)/2$. Neste caso, a união das duas curvas é contínua em t_1 , tanto em posição ($p(t_1) = q(t_1)$) quanto velocidade ($p'(t_1) = q'(t_1)$).

A mesma condição garante a junção suave (contínua e com derivada contínua) de dois polinômios de Bézier $r(t)$, $s(t)$ definidos sobre intervalos iguais $[t_0 - t_1]$ e $[t_1 - t_2]$

Parte prática. Escreva e teste as macros abaixo, conforme especificado na classe:

- `inicializa_spline_3(N,C,uu,vv,aa,bb)`: Supõe que N é um inteiro e que o parâmetro C é um array de tamanho $[N] [4]$ que vai definir uma curva ou polinômio de Bézier formada de N segmentos. Preenche esse array com os pontos ou valores de controle de N segmentos de Bézier, escolhidos aleatoriamente em função dos parâmetros uu, vv, aa, bb como será explicado em classe.
- `suaviza_spline_3(N,C)`: Supõe que os parâmetros N e C são como no item anterior. Modifica os valores de controle em C de modo que os segmentos emendem suavemente.
- `extrai_spline_3(T,k,N,C)`: Supõe que T é um array de 3 índices $[M] [N] [4]$, definindo M splines de Bézier, e N e C são como no item anterior. Copia $T[k] [i] [j]$ em $C[i] [j]$, para i, j em $0..N-1, 0..3$.
- `guarda_spline_3(T,k,N,C)`: Supõe que T, N e C são como no item anterior. Copia $C[i] [j]$ em $T[k] [i] [j]$, para i, j em $0..N-1, 0..3$.

- `define_tentaculos_3(M,N,P,R)`: Supõe que `P` é um array de pontos e `R` é um array de números, ambos com dimensões `[M] [N] [4]`. Preenche esses arrays com os dados (curvas-eixo e raios) dos tentáculos do seu monstro.
- `gera_tentaculos_3(nb,M,N,P,R,...)`: cria modelo POV-Ray para esse tentáculo, usando `nb` bolas em cada segmento.
- `gera_monstro(m,M,N,...)`: gera o monstro com `M` tentáculos de `N` segmentos cada, com `nb` bolas em cada segmento.

Observações. O arquivo de descrição `main.pov` deve ser construído manualmente, com um editor de texto comum, **sem** o auxílio de qualquer editor gráfico ou outra ferramenta de modelagem geométrica. Não é permitido copiar ou incluir quaisquer arquivos POV-Ray além dos fornecidos pelo professor ou escritos por você mesmo, neste exercício ou em exercícios anteriores.

Lembre-se de que todo trabalho prático é **individual**. **Não se esqueça de executar o comando `make export` até o final da aula.**