

MC937A/MO603A – Computação Gráfica - 2023-S2 - Jorge Stolfi
Trabalho de laboratório 06 - 2023-09-20
Grafo desorientado (III)

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

Objetivos. Treinar o conceito de splines de Bézier.

Enunciado. Nesta aula vamos completar a obfuscação, digo, visualização de um grafo, usando *splines* de Bézier para modelar as arestas. Um *spline*, de modo geral, é uma curva suave definida por uma seqüência de pontos. Um *spline de Bézier* é um spline produzidas pela matro `interpola3_multi` desenvolvida no laboratório passado.

Macro principal: A macro principal será `gera_grafo(nv,ne,org,dst,D,na,nb)`, onde `nv` é o número de vértices, `ne` é o número de arestas, e `org,dst` são duas *arrays* de `ne` elementos, tais que a aresta `ie` vai do vértice de índice `org[ie]` para o vértice de índice `dst[ie]`. Veja exemplo em classe.

A macro deve calcular internamente o grau `deg[iv]` de cada vértice $iv \in \{0..nv - 1\}$. O grafo deve ser tal que nenhum vértice tem grau maior que 4.

Gerando os vértices: A macro `gera_grafo` deve criar `nv` objetos que representam os vértices. As três coordenadas de cada vértice devem ser escolhidas aleatoriamente dentro de uma região cúbica de lado `D`.

Como nas aulas passadas, o objeto que representa um vértice `iv` de grau $k = \text{deg}[iv]$ deve ter k *terminais* — pontos específicos na sua superfície onde os cordões que representam essas k arestas entram no objeto. Cada terminal é definido por dois pontos, uma *base* na superfície do objeto e uma *ponta* fora dele.

Conectando os terminais: Use as *arrays* `org,dst` para decidir que vértices devem ser ligados por cada aresta, e um *array* de contadores `nt[0..nv-1]` para saber quantos terminais de cada vértice já foram usados. Cada par de terminais deve ser conectado usando um cano curvo, formado por esferas cujos centros percorrem a curva definida por `interpola3_multi`.

Mais especificamente, escreva uma macro `conetor(na,nb,pini,tini,pfin,tfin)` que gera tal cano, dados a base `pini` e a ponta `tini` de um terminal, e a base `pfin` e a ponta `tfin` do outro terminal. O conector deve consistir de `na` arcos de Bézier, cujos pontos de controle `P1[0..na-1]` e `P2[0..na-1]` devem ser declarados e definido dentro da macro. Inicialmente, use coordenadas aleatórias para os pontos de controle, exceto `pini,tini,pfin,tfin`. A macro `conetor` deve gerar `nb` bolas em cada arco da curva.

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.