

# MAC 420/5744 – Introdução à Computação Gráfica

Prof. Marcel Parolin Jackowski

BCC - IME/USP – Primeiro Semestre de 2015

## Terceiro Exercício-Programa

Data de entrega: até 4 de julho de 2015

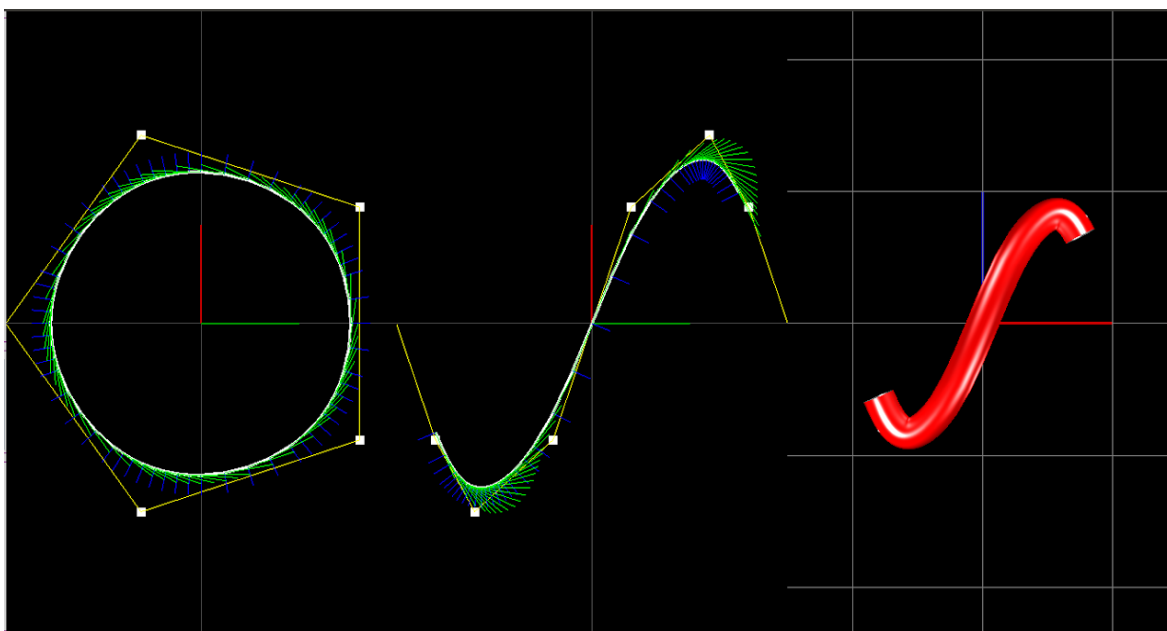
### Modelagem Geométrica com RaGs e B-splines

#### 1. Introdução

As curvas e superfícies B-splines apresentam uma forma flexível de representação geométrica, dispondo de controle local e de continuidade adequada para o desenho de formas curvilíneas. Já as curvas e superfícies racionais gaussianas (RaGs) são uma alternativa às B-splines, pois adicionam um grau adicional de liberdade em sua representação: o desvio padrão da função Gaussiana, que controla a suavidade da curva ou superfície resultante. Neste EP você exercitará o estudo e implementação de curvas B-spline e RaG para criar um software de modelagem geométrica baseada na extrusão de formas bidimensionais utilizando WebGL.

Inicialmente familiarize-se com B-splines lendo o livro-texto e logo após lendo o artigo sobre curvas e superfícies racionais gaussianas de A. Goshtasby, "Geometric modelling using rational Gaussian curves and surfaces", *Computer-Aided Design*, v. 27:5, pp. 363-375, 1995, disponível através do link:

<http://www.sciencedirect.com/science/article/pii/0010448595968002#>.



A figura acima demonstra a funcionalidade esperada da sua aplicação, que deverá conter 3 canvases. Um deles (a esquerda) serve para o desenho de uma curva fechada (perfil) utilizando pontos de controle escolhidos pelo usuário, o outro (meio) serve para o desenho de uma outra curva, neste caso aberta, que define a trajetória da varredura. O último canvas (a direita) mostra o resultado da extrusão em 3D, que é o resultado da varredura da forma fechada ao longo da segunda curva que define a trajetória. Você poderá dispor estes canvases da forma que mais achar apropriada. Você tem a liberdade de mudar a cor de fundo, das curvas e da superfície resultante. Você não precisará desenhar as retas tangentes (verde) ou normais (azul) conforme ilustradas na figura.

## 2. Requisitos

Utilizando o botão esquerdo do mouse, você deverá ser capaz de adicionar pontos de controle ou movimentá-los caso já existam nas posições clicadas pelo mouse (utilize um valor de tolerância para determinar se um ponto já existe naquela posição). Este comportamento deve funcionar nos dois canvases 2D. Os pontos deverão ser desenhados como pequenos círculos ou quadrados conforme ilustração acima.

Você deverá prover meios de limpar cada canvas para inicializar um novo desenho. Tanto as curvas RaG quanto B-splines precisam de número de subdivisões para serem desenhadas. Você adicionará um *widget* na interface onde o usuário poderá escolher o número de pontos da curva a serem calculados. Os pontos de cada curva deverão ser conectados através de segmentos de reta (poli-linhas). O usuário também deverá ter a liberdade para escolher o tipo de representação (RaG ou B-spline), seja para desenhar o perfil, ou para desenhar a trajetória. A mudança de representação não deverá apagar os pontos de controle previamente inseridos, mas simplesmente redesenhar uma nova curva de acordo com a representação escolhida. Note que RaGs precisam do valor do desvio padrão (e.g. 0.0-1.0 em incrementos de 0.01) e as B-splines necessitam do grau do polinômio (0-n, n número inteiro). Você deverá adicionar *widgets* apropriados para a entrada destes valores. Desenhe a curva resultante e o polígono de controle.

No canvas 3D, você desenhará uma malha triangular que representará a extrusão do perfil ao longo da trajetória desenhada. Para tanto, você irá replicar os pontos da curva perfil ao longo das subdivisões da curva que representa a trajetória. Você poderá assumir que a origem do sistema de coordenadas do canvas perfil seja o centro pelo qual a extrusão por varredura ocorrerá. Note que você precisará calcular o vetor tangente da trajetória para orientar corretamente a curva perfil ao longo do caminho. O número de triângulos da malha será proporcional ao número de subdivisões do perfil e da trajetória. A superfície deverá aparecer no centro do seu canvas 3D e o usuário poderá utilizar o trackball virtual para rotacionar a cena. Determine uma escala adequada para sempre mostrar toda a superfície resultante. Você precisará calcular as normais para cada vértice e poderá deduzir os vetores normais diretamente das representações. Adicione uma fonte de luz local que favoreça o reconhecimento tridimensional da forma resultante.

### 3. Implementação

Capriche na apresentação e na usabilidade do seu software - você está cursando Computação Gráfica! Você só poderá utilizar HTML5, WebGL e Javascript. Não será permitida nenhuma versão compilada ou traduzida a partir de outra linguagem. Não será permitido o uso de qualquer biblioteca ou módulo externo. Siga os requisitos rigorosamente, e quando em dúvida sempre consulte o monitor da disciplina ou o professor.

### 4. Avaliação

#	Descrição	Porcentagem
1	Correta implementação e desenho de curvas B-spline	20 %
2	Correta implementação e desenho de curvas RaG	20 %
3	Correta adição e movimentação de pontos de controle	10 %
4	Controle do grau de subdivisão de cada curva	5 %
5	Parâmetros sigma e grau de polinômio em funcionamento	15 %
6	Trackball em funcionamento	5 %
7	Renderização adequada da superfície por extrusão	15 %
	<b>Total</b>	<b>100 %</b>

### Observações gerais

**Este será um EP feito em até duas pessoas.**

- Você poderá apenas utilizar as funções da biblioteca Common, mas nenhuma outra biblioteca.
- LEIAME.TXT contendo instruções ou comentários sobre o seu software
- Entregue um único arquivo através do PACA, compactado (.tgz ou .zip) contendo o código-fonte, e com os nomes dos participantes. Ex: Marcel-Miguel.tgz.