

Trabalho 3 - Visualização de uma superfície B-Spline

Descrição:

Desenvolva um programa em C++, que gere uma malha de triângulos baseada em uma superfície B-Spline. O usuário deve poder rotacionar a malha nos eixos x, y e z. Pode ser desenvolvido na Canvas2D ou na API SCV.

Requisitos:

- Gerar uma malha triangular baseada em uma superfície B-Spline com 10x10 pontos de controle tridimensionais previamente definidos (ver figuras 1 e 2). Esses pontos podem ser gerados randomicamente (ou pseudo-randomicamente – função procedural);
- O usuário deve poder controlar o refinamento da malha (número de triângulos) através de um slider;
- O usuário deve poder rotacionar a malha (nos eixos x, y e z) para que a mesma possa ser visualizada em três dimensões;
- Definir câmera em perspectiva.

Extras:

- Animação procedural da malha para, por exemplo, gerar ondas;
- Uso estruturado de matrizes de transformação;
- Etc.

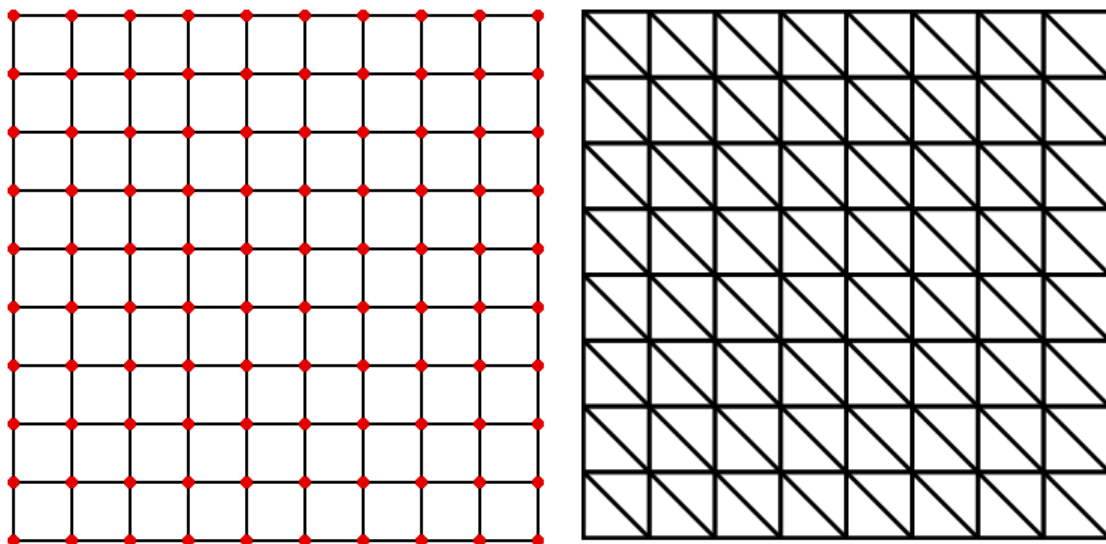


Figura 1. Pontos de controle e malha triangular.

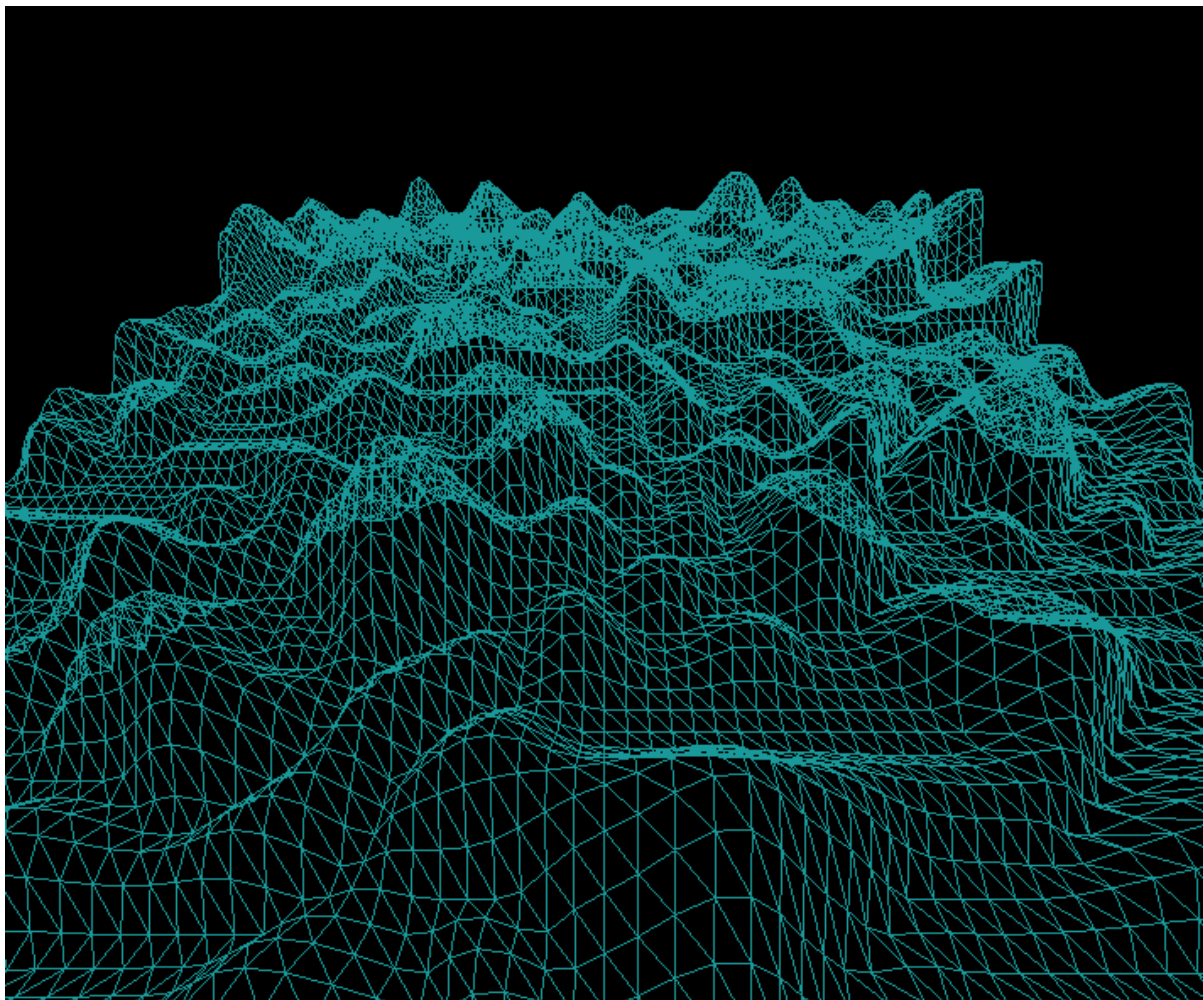


Figura 2. Terreno com altura variável gerado através de malha de triângulos.

Data e Formato de Entrega

- Data: 15/06/2015
- No email e no cabeçalho do arquivo, devem conter o nome completo e matrícula do aluno.
- O arquivo deve ser enviado para pozzer3@gmail.com, gcardozo@inf.ufsm.br e afrasson@inf.ufsm.br com o subject "CG T3".
- Deve-se enviar fontes e o projeto para o compilador **Visual Studio 2013**. Envie **apenas** os arquivos **.sln** e **.vcxproj** juntamente com a pasta **src**.
- O programa deve ser enviado em um arquivo compactado fulano.rar (fulano = login ou nome do aluno). Dentro deste arquivo deve haver um diretório com o mesmo nome do arquivo e dentro deste diretório os arquivos do trabalho.
- Ex: o arquivo pozzer.rar deve conter um diretório chamado pozzer, e dentro do diretório devem estar os arquivos do trabalho.

Critério de Avaliação

- Documentação: descrever no cabeçalho de cada arquivo a ideia geral do código e detalhes específicos de partes que mereçam uma explicação – não comente por exemplo o que faz b++.
- README.txt: incluir um arquivo “README.txt” contendo informações sobre quais funcionalidades foram implementadas (requisitos e extras).
- Pontualidade: Trabalhos não entregues na data não serão avaliados e receberão nota zero.
- Legibilidade: nome de variáveis, estruturação do código. O código digital a ser entregue deve ter 4 espaços de indentação e não deve possuir tabulações.
- Clareza: facilidade de compreensão – evite códigos complexos e desnecessários. Adote a solução mais simples possível.
- Funcionalidade: o programa deve satisfazer todos os requisitos. Programas que não compilarem ou que não atenderem nenhum requisito receberão nota 0 (zero).

Você pode discutir estratégias e ajudar o colega na implementação, porém evite passar código fonte. Programas semelhantes terão a nota 0 (zero).