Tópicos em Geometria Computacional

Primeiro Trabalho

24 de maio de 2021

1 Introdução

O trabalho consiste em fazer uma implementação do problema *triangulação de polígonos*, descrito na Seção 6. Neste problema é preciso implementar funções para manipulação de uma estrutura de dados para representar triangulações.

2 Resolução do problema

A resolução do problema, ou seja, a descrição do problema, do algoritmo e sua corretude, deve estar em um texto claro em formato de um artigo e em pdf (com nome relatorio.pdf). Deve conter o nome do autor (aluno), uma introdução com o problema, o algoritmo e sua explicação. Todas as referências que forem usadas devem estar citadas corretamente no texto.

3 Especificação da implementação

A implementação pode ser feita em qualquer linguagem, contanto que seja possível rodar no ambiente computacional do DINF.

A entrada de dados deve ser feita pela entrada padrão (stdin) e a saída de dados pela saída padrão (stdout), ou seja, o seu programa lê do teclado e escreve na tela. O objetivo é que seja executado com redirecionamento de arquivos, como o comando abaixo:

$ triangulate < entrada.txt > saida.txt

O trabalho deve ser entregue com um makefile de forma que ao digitar o comando make o executável seja construído.

4 Entrega

A entrega deve ser feita por e-mail para `andre@inf.ufpr.br`, com um arquivo tar.gz (compactado) enviado como anexo de uma mensagem com assunto "geocomp-trabalho 1" (exatamente).

O arquivo compactado deve ter nome fulano.tar.gz, onde fulano deve ser substituído pelo "login name" do autor.

O arquivo fulano.tar.gz, uma vez expandido, deve criar (no diretório corrente) os arquivos abaixo:

- makefile (ou Makefile)
- relatorio.pdf
- e os arquivos fontes necessários para a geração do executável.

5 Estrutura de Dados para Triangulações

Uma estrutura de dados para representar uma triangulação no plano pode ser a descrita abaixo.

Cada vértice recebe um índice, de 1 a n, e suas coordenadas são armazenadas em um vetor indexado por estes índices.

Cada triângulo recebe um índice, de 1 a m, e um vetor indexado por estes índices deve conter: três campos para os vértices do triângulo, V_1, V_2, e V_3, com os índices dos três vértices; três campos com os índices dos triângulos vizinhos, T_1, T_2 e T_3, de forma que o triângulo T_i seja oposto ao vértice V_i.

Os vértices de um triângulo devem estar em ordem horária e o primeiro índice (V_1) deve ser o menor.

Caso a face externa não seja um triângulo, esta recebe o índice 0.

6 Triangulação de Polígonos

Neste problema o objetivo é encontrar uma triangulação de um polígono dado. O nome do programa executável deve ser triangulate.

A entrada de dados será um texto com a descrição do polígono. A primeira linha tem um número, n, de vértices do polígono. As n linhas seguintes tem as coordenadas X e Y de cada um dos vértices, separadas por um espaço em branco. Os vértices do polígono podem estar em ordem horária ou anti-horária, ou seja, não existe uma padronização.

Exemplo:

```
4
1 4
1 20
15 20
15 4
```

A saída de dados deve ser um texto com a descrição da estrutura de dados que representa a triangulação encontrada. As primeiras linhas são iguais a entrada, com a descrição dos vértices da triangulação (que são os mesmos do polígono). Após estas primeiras linhas, temos uma linha com o número de triângulos e depois uma linha para cada triângulo, com os 6 campos da estrutura de dados, separados por um espaço em branco.

Exemplo:

```
4
1 4
1 20
15 20
15 4
2
1 2 3 0 2 0
1 3 4 0 0 1
```