

Modelagem Computacional em Grafos

SCC0216

Disponibilizado em: 27-02-2023

Código Turma A (terça): **6NGE**

Submissões até: 15-04-2023

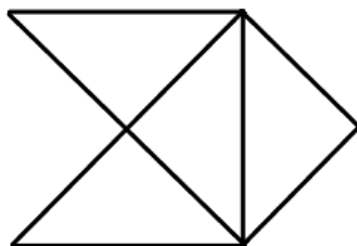
Código Turma B (segunda): **1KNW**

Desenhos Eulerianos

1. Introdução

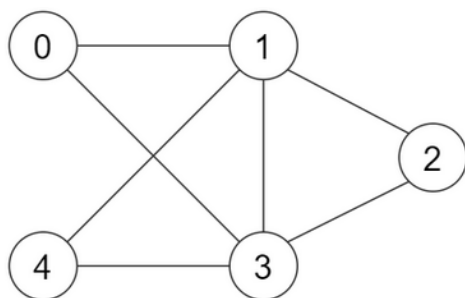
Bento, como um bom estudante, estava assistindo as aulas de Cálculo 3 do professor Possani para tentar entender alguma coisa sobre integrais. Vendo as aulas, percebeu que seria necessário aprender a desenhar diferentes regiões de integração e ficou levemente preocupado, já que não tinha habilidade nenhuma para isso.

Motivado pela vontade de se livrar logo da disciplina, resolveu aprender a desenhar e começou com as formas mais simples, como retângulos e triângulos. Em determinado momento, ao combinar as formas chegou ao desenho abaixo:



Curiosamente, Bento percebeu que para este desenho, iniciando em um determinado ponto, não precisou tirar o lápis do papel ou retroceder os traços, e traçando cada linha uma única vez retornou ao ponto inicial. Empolgado com sua observação ele foi logo contar para você.

Você como um bom computeiro, entendido de grafos, prontamente contou que a maneira que ele traçou o desenho pode ser entendida como um circuito euleriano no grafo euleriano abaixo:



Como você é um bom amigo resolveu, juntamente com Bento (que de tão empolgado esqueceu totalmente de Cálculo 3!), criar um programa que verifica para diferentes desenhos se é possível traçá-los da mesma forma que foi feito anteriormente e qual a trajetória a ser seguida para obtê-los, ou seja, dado um grafo conexo verifica se ele é euleriano, e se for apresenta o circuito euleriano no grafo.

2. Entrada

A entrada para seu programa é um grafo conexo, que representa o desenho. Seu programa deverá ler da entrada padrão:

- Uma string, que é o nome de um arquivo contendo as seguintes informações, separadas por linha:
 - Dois inteiros: v e a , que representam respectivamente a quantidade de vértices e arestas do grafo.
 - a linhas que informam dois inteiros, que representam os vértices extremos ligados por uma aresta

Observação: os vértices são necessariamente rotulados como inteiros de 0 a $(v - 1)$, como no exemplo, em que $v = 5$.

Ou seja, se temos, por exemplo, um arquivo que representa o grafo mostrado anteriormente, ele será dessa forma:

```
5 7
0 1
0 3
1 2
1 3
1 4
2 3
3 4
```

3. Saída

A saída esperada do seu programa é:

- A string “Sim” caso seja possível traçar o desenho da forma descrita (ou seja, o grafo possui circuito euleriano); ou
- A string “Não” caso não seja possível traçar o desenho da forma descrita.
- Caso tenha impresso “Sim”, na linha seguinte vem a sequência de vértices (inteiros) que corresponde ao circuito euleriano, separados por um espaço em branco.

4. Orientações para a confecção do código

Ao buscar por um circuito euleriano, **inicie sempre pelo vértice representado pelo valor 0**, e ao procurar pelo caminho no grafo, **ao escolher uma aresta a incluir no caminho entre várias opções possíveis, escolha sempre a aresta que leva ao vértice de menor valor**. Por exemplo, no grafo acima, o circuito seria indicado por 0 1 2 3 1 4 3 0

Na entrada, será fornecido o nome do arquivo

5. Exemplos de entrada e saída

Entrada: exemplo.in	Saída: Sim 0 1 2 3 1 4 3 0
-------------------------------	---

Arquivo exemplo.in 5 7 0 1 0 3 1 2 1 3 1 4 2 3 3 4

Entrada: 2.in	Saída: Sim 0 1 2 0 5 2 3 4 5 6 0
-------------------------	---

Arquivo 2.in 0 1 0 2 0 5 0 6 1 2 2 3 2 5 3 4 4 5
--

Bom Trabalho :)