

Tarefa Nº 02 - Testes em Circuitos Integrados

Prazo de entrega: **Consultar a página da tarefa.**

Linguagem para implementação: **C++**.

Professor: Andrei Braga

ATENÇÃO: Para a realização desta tarefa, você deve utilizar **apenas** os conceitos e algoritmos estudados até a aula do dia **28/04/2023** (Aula 04).

Testes em Circuitos Integrados

O prof. Caimi está selecionando circuitos integrados para serem utilizados em uma atividade com os seus alunos. Um circuito integrado será adequado para a atividade se satisfizer a determinadas condições. Como são muitos circuitos a serem avaliados, o prof. Caimi pediu que você use os seus conhecimentos de grafos para ajudá-lo a escrever um programa para realizar os testes necessários.

Um circuito integrado pode ser modelado como um grafo onde os vértices representam as portas lógicas do circuito e as arestas representam as conexões entre estas portas lógicas. Os testes que o prof. Caimi precisa realizar correspondem às seguintes operações em um grafo:

- operação `eh_caminho`: dada uma sequência de vértices do grafo, verifica se a sequência é um caminho;
- operação `existe_caminho`: dados dois vértices do grafo, verifica se existe um caminho entre os dois vértices;
- operação `eh_conexo`: verifica se o grafo é conexo;
- operação `eh_aciclico`: verifica se o grafo é acíclico (ou seja, não possui ciclos).

Nesta tarefa, você deve implementar uma classe que represente um grafo **simples**. Você deve fazer isso escrevendo métodos que devem executar no grafo as operações listadas acima.

Você deve escrever um programa que constrói um grafo, executa operações no grafo e depois, se necessário, explicitamente o destrói. O seu programa deve processar informações que determinarão as operações a serem executadas no grafo, o que deve ser feito de acordo com as **Seções Entrada e Saída** abaixo.

Entrada

A primeira linha da entrada contém dois inteiros **V** ($V > 0$) e **E** ($E \geq 0$), sendo **V** o número de vértices do grafo a ser construído e **E** o número de arestas a serem inseridas no grafo. Cada uma das **E** linhas seguintes contém dois inteiros **X** e **Y**, indicando que a aresta **X Y** deve ser inserida no grafo.

A próxima linha da entrada contém um inteiro **O** ($O \geq 0$), que determina a quantidade de operações a serem executadas no grafo. Cada uma das **O** linhas seguintes consiste em uma das opções abaixo:

- O caractere S, um inteiro **N** e **N** inteiros separados por espaços em branco;
- O caractere C, um inteiro **X** e um inteiro **Y** separados por espaços em branco;
- O caractere X;
- O caractere A.

Estas opções representam o seguinte, de acordo com o primeiro caractere da linha:

- Se é S, então, após **N**, é dada uma sequência de **N** vértices do grafo. Deve ser executada a operação de verificar se a sequência dada é um caminho (método `eh_caminho`);
- Se é C, então deve ser executada a operação de verificar se existe um caminho entre os vértices **X Y** (método `existe_caminho`);
- Se é X, então deve ser executada a operação de verificar se o grafo é conexo (método `eh_conexo`);
- Se é A, então deve ser executada a operação de verificar se o grafo é acíclico (método `eh_aciclico`).

Saída

Para cada execução de uma das operações descritas acima, o seu programa deve imprimir o seguinte:

- caso o resultado da verificação seja verdadeiro, uma linha contendo o texto
SIM
- caso contrário, uma linha contendo o texto
NAO

Exemplos de execução

Entrada	Saída
4 3 0 1 1 2 2 3 6 S 3 1 2 3 S 4 2 1 0 1 S 3 0 3 2 C 2 0 X A	SIM NAO NAO SIM SIM SIM SIM

Entrada	Saída
4 5 0 1 0 2 1 2 1 3 2 3 5 C 3 0 S 2 2 0 S 3 2 0 3 X A	SIM SIM NAO SIM NAO

Observações:

- Para a realização dos testes automáticos, a compilação se dará da seguinte forma:
`g++ -pedantic -Wall *.cpp -lm -lutil`