

Este projeto consiste em implementar duas versões diferentes de algoritmos para verificar se um grafo é conexo ou não.

### Buscas e Conexidade

Conforme apresentado em aula, os algoritmos de buscas em grafos (busca em largura e busca em profundidade) podem ser usados, também, para verificar se um grafo não orientado é conexo ou não.

Na sua implementação, você deverá:

- Implementar o algoritmo de busca em largura conforme publicado no livro do Cormen (pag 595, na 3a. edição em inglês); tal implementação deverá ser feita através de uma função específica para isto e toda interface dela deverá ser feita através de argumentos e, eventualmente, valores retornados.
- Implementar o algoritmo de busca em profundidade conforme publicado no livro do Cormen (pag 604, na 3a. edição em inglês); tal implementação deverá ser feita através das duas funções especificadas no livro e toda interface delas deverá ser feita através de argumentos e, eventualmente, valores retornados.

Além de implementar os algoritmos das buscas, você deverá:

- Implementar uma função chamada **eConexoBLargura** que retornará **1**, se o grafo for conexo, ou **0**, caso contrário; tal função deverá usar o algoritmo de busca em largura implementado e, usando os detalhes calculados pelo algoritmo de busca, decidir se o grafo é conexo ou não.
- Implementar uma função chamada **eConexoBProf** que retornará **1**, se o grafo for conexo, ou **0**, caso contrário; tal função deverá usar o algoritmo de busca em profundidade implementado e, usando os detalhes calculados pelo algoritmo de busca, decidir se o grafo é conexo ou não.

### Implementação de Grafo

Sua implementação deverá, obrigatoriamente, usar a implementação de grafos em C publicada no Moodle, sem fazer nenhuma adaptação na estrutura de dados adotada.

Será permitido, apenas, ajustes nos atributos dos vértices para incluir os detalhes manipulados pelos algoritmos de buscas (conforme apresentados no livro do Cormen).

### Resultado Esperado

Seu programa deverá exibir, além do resultado referente à conexidade do grafo, os atributos calculados pelos algoritmos de busca (para cada vértice do grafo) Implemente as impressões dos resultados através de 2 funções que, individualmente, imprimam os detalhes calculados em cada uma das buscas.

Implemente uma função **main** que inicializa um único grafo com um valor constante e calcula sua conexidade através das duas versões das funções solicitadas.

### Observações Complementares

1. O trabalho pode ser desenvolvido por grupos de até 5 alunos.
2. Um único aluno do grupo deverá publicar o trabalho no Moodle (penalidade: 2,0 pontos).
3. Deverá ser entregue um único arquivo, com extensão .c, com todo o código na linguagem C (padrão ANSI C).
4. O arquivo deverá conter um cabeçalho (comentário) com as identificações completas de todos os membros do grupo (penalidade: 2,0 pontos).
5. Documentar adequadamente o programa e incluir comentários úteis e informativos.
6. Não usar variável global no programa (penalidade: 2,0 pontos).
7. Seu programa será testado no DEV para Windows.
8. Entrega deverá ser feita até as 18:00 do dia 18 de novembro.