

CI1065 - Algoritmos e Teoria dos Grafos

Primeiro Trabalho

7 de junho de 2023

1 Introdução

O trabalho consiste implementar um algoritmo para o problema da *Ordenação Topológica* de grafos direcionados acíclicos. A resolução do problema, ou seja, uma breve descrição do problema, da modelagem e da implementação, deve estar em um texto claro em formato de um artigo em pdf. O texto deve conter o nome dos autores (alunos), uma introdução com o problema e os detalhes da implementação (com exemplos de uso). Todas as referências que forem usadas devem estar citadas corretamente no texto.

Você pode usar bibliotecas para estruturas de dados (como listas, conjuntos etc) e para leitura de dados da entrada, mas não para o algoritmo de resolução principal do problema. O seu programa deve compilar e executar nas servidoras do DINF.

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

Resumindo, o texto deve ter:

- identificação;
- explicação do problema;
- detalhes da implementação.

Você deve entregar um arquivo compactado (`tar.gz`) com os seguintes arquivos no diretório corrente:

- texto (em pdf);
- os fontes (podem estar em subdiretórios);
- `makefile`;
- exemplos usados (podem estar em subdiretórios).

A entrega deve ser feita por e-mail para `murilo@inf.ufpr.br` em um arquivo compactado com todos os arquivos do trabalho, com assunto “Grafos - Trabalho 1” (exatamente).

2 O problema: Ordenação Topológica

Uma *ordenação topológica* de um grafo direcionado G é uma permutação (v_1, \dots, v_n) de $V(G)$ que “respeita a direção dos arcos” de G , isto é,

$$i < j, \text{ para todo } (v_i, v_j) \in A(G).$$

O reverso da pós-ordem de uma floresta direcionada resultante de uma busca em profundidade em um grafo direcionado acíclico G é uma ordenação topológica de G .

Neste trabalho você deve implementar um algoritmo que toma como entrada um grafo direcionado acíclico e retorna os seus vértices na ordem topológica.

2.1 Formato de entrada e saída

Os formatos de entrada e saída, são descritos a seguir e devem ser usados a entrada e a saída padrão (`stdin` e `stdout`). Para a representação de grafos na entrada e saída ao longo da disciplina vamos utilizar a linguagem de descrição de grafos `dot` que é implementada pelo pacote de software `GraphViz`.

Entrada: Um grafo direcionado acíclico (GDA) no formato `dot`.

Saída: Arquivo texto com uma linha com os vértices do grafo ordenados topologicamente, separados por espaço (simples) e sem espaço no começo nem no fim da linha. Se o grafo de entrada não for um GDA, o programa deve retornar uma mensagem de erro.