

PUC-Rio
Departamento de Informática
Prof. Marcus Vinicius S. Poggi de Aragão
Horário: 2as. e 4as. 13-15hs
28 de maio de 2022
Entrega: 23:59hs 18 de junho de 2022
Período: 2022.1

ANÁLISE DE ALGORITMOS (INF 1721)

3º Trabalho de Implementação

Descrição

Este trabalho prático consiste em desenvolver códigos para diferentes algoritmos e estruturas de dados para resolver os problemas descritos abaixo e, principalmente, analisar o desempenho das implementações destes algoritmos com respeito ao tempo de CPU. O desenvolvimento destes códigos e a análise experimental devem seguir os seguintes roteiros:

- Descrever os algoritmos informalmente.
- Demonstrar o entendimento do algoritmo explicando, em detalhe, o resultado que o algoritmo deve obter e justificá-lo.
- Explicar a fundamentação do algoritmo e justificar a sua corretude. Apresentar e explicar a complexidade teórica esperada para cada algoritmo.
- Documente o arquivo contendo o código fonte de modo que cada passo do algoritmo esteja devidamente identificado e deixe claro como este passo é executado.

A corretude código será testada sobre um conjunto de instâncias que será distribuído. O trabalho entregue deve conter:

- Um documento contendo o roteiro de desenvolvimento dos algoritmos (e dos códigos), os itens pedidos acima, comentários e análises sobre a implementação e os testes realizados (pdf)
- Código fonte (todos os arquivos para gerar o executável ou o iPython notebook).
- Recomenda-se a entrega do trabalho em Jupyter Notebook(Python, Julia).
- Um e-mail contendo os códigos fonte e os executáveis correspondentes deve ser enviado para **poggi@inf.puc-rio.br**.

- É **OBRIGATÓRIO** o uso do ASSUNTO (ou SUBJECT) AA221T3-GG, onde GG é o número do grupo.
- A falta do e-mail COM este ASSUNTO implica na **NÃO** consideração do trabalho e nota **Zero!**
- O trabalho pode ser feito em grupos de 2 (dois) a 4 (quatro) alunos.
- Os arquivos com os dados para os problemas são disponibilizados no site do curso.

Avaliação

O trabalho deverá conter também uma seção de avaliação. Esta seção conterá a nota que o grupo atribui para o trabalho (0-10) e uma justificativa de até 1000 caracteres.

Cada membro do grupo GG deverá enviar um email para **poggi@inf.puc-rio.br**, também com o assunto AA221T3-GG, com a auto-avaliação da sua participação no trabalho e as avaliações das contribuições dos demais membros do grupo. Em todos os casos, além do grau (0-10), o aluno poderá justificar utilizando até 500 caracteres.

Importante: Na avaliação de cada membro do grupo, **indicar o nome e o último sobrenome** do membro avaliado.

Novamente, o não envio deste e-mail com este *assunto* implicará na nota **Zero**.

1 Problema do Fluxo Máximo: *Preflow-Push Algorithms*

Fluxo Máximo (FM) - Dado um grafo orientado $G = (V, A)$, capacidades positivas u_a associadas aos arcos $a \in A$ e vértices $s, t \in V$. Determinar o fluxo $f = (f_a, a \in A)$ tal que $0 \leq f_a \leq u_a, a \in A$, e $\sum_{a \in \delta^-(v)} f_a = \sum_{a \in \delta^+(v)} f_a, v \in V - \{s, t\}$, que maximiza:

$$z = \sum_{a \in \delta^+(s)} f_a = \sum_{a \in \delta^-(t)} f_a$$

Implemente e faça uma comparação com as respectivas complexidades teóricas. Não é preciso apresentar a prova da complexidade teórica, mas é preciso considerar a complexidade conhecida do algoritmo. A ênfase do trabalho deve ser numa minuciosa análise experimental destes algoritmos sobre os conjuntos de instâncias disponibilizados (arquivo maxflow-instances-zuse88.zip na pasta AA221-T3 no Google Drive do curso). Algoritmos:

1. Algoritmo *Edmonds-Karp*, i.e., algoritmo de Ford-Fulkerson utilizando Busca em Largura para determinar se existe caminho $s - t$ na rede residual corrente. Referência: AMO Seção:6.4.

2. Algoritmo *Preflow-Push* genérico. Referência: AMO Seção:7.6
3. Algoritmo *FIFO Preflow-Push*. Referência: AMO Seção:7.7
4. Algoritmo *Highest Label Preflow-Push*. Referência: AMO Seção:7.8

AMO: R.K. AHUJA, T.L. MAGNANTI e J.B. ORLIN, *Network Flows*, Prentice Hall, 1993.