

Atividade A3 – Grafos (INE5413)
Ciências da Computação – Universidade Federal de Santa Catarina
Prof. Rafael de Santiago

Observações gerais:

- Trabalho deve ser executado em no máximo 3 estudantes da disciplina.
- Todas as codificações devem estar em uma das seguintes linguagens de programação: C/C++, Python ou Java.
- Todos os programas devem ler o padrão das instâncias disponibilizadas pelo professor no MOODLE para o referido problema.
- A entrega do código-fonte deverá ser realizada no MOODLE usando a plataforma VPL intitulada “Atividade A3 - Entrega dos códigos-fonte/programas”^a.
- A entrega do relatório é obrigatória. O relatório deve estar no formato PDF. A entrega do relatório deverá ser realizada no MOODLE via recurso intitulado “Atividade A3 - Entrega do relatório”.
- Duas ou mais equipes com trabalhos total ou parcialmente iguais receberão nota 0.
- Instruções gerais para uso do VPL no MOODLE:
 - O ambiente de teste foi preparado para as questões 1 a 3 da atividade A3;
 - Um programa deve ser preparado para cada questão da atividade, contendo o nome do arquivo “**A3.X.ext**”, no qual **X** é o número da questão e **ext** é a extensão do código-fonte na linguagem de programação aceita. Por exemplo, **A3_2.py** é o arquivo que o estudante pode ter preparado para testar a segunda questão da atividade A3, utilizando a linguagem Python.
 - Os programas devem estar preparados para receber os parâmetros via chamada de execução no console/terminal na ordem estabelecida no enunciado de cada questão. A título de exemplo, o ambiente de teste iria executar a questão 2 com o seguinte comando (solicitando o emparelhamento máximo no grafo “pequeno.net”): **python3 A3_2.py pequeno.net**.
 - A saída deve respeitar o formato estabelecido para cada questão no enunciado da atividade.
 - É possível adicionar arquivos auxiliares (pacotes/bibliotecas produzidas pelo estudante).

^aA entrega deve ser realizada através do ambiente da turma no MOODLE.

1. [Edmonds-Karp] (2,5pts) Crie um programa que receba um grafo dirigido e ponderado como argumento o vértice de origem s e o vértice de destino t . Ao final, imprima na tela o valor do fluxo máximo resultante da execução do algoritmo de Edmonds-Karp considerando o fluxo de s para t , como no exemplo abaixo.

17

2. [Hopcroft-Karp] (2,5pts) Crie um programa que receba um arquivo de grafo bipartido, não-dirigido, não-ponderado e informe qual o valor do emparelhamento máximo e quais arestas pertencem a ele. Utilize o algoritmo de Hopcroft-Karp. Ao final, imprima na tela a quantidade de emparelhamentos encontrados (na primeira linha) e quais são as arestas correspondentes (na segunda linha), como no exemplo abaixo. Para as instâncias disponibilizadas pelo professor, considere que para um grafo com n vértices, os vértices $1, 2, \dots, \lfloor \frac{n}{2} \rfloor$ estão na primeira parte e os vértices $\lfloor \frac{n}{2} \rfloor + 1, \lfloor \frac{n}{2} \rfloor + 2, \dots, n$ estão na segunda parte do grafo bipartido.

3

1-4, 2-5, 3-6

3. [Coloração de Vértices] (2,5pts) Crie um programa que recebe um grafo não-dirigido e não-ponderado como argumento. Ao final, informe a coloração mínima e qual número cromático foi utilizado em cada vértice. Use o algoritmo de Lawler para cumprir essa questão. Ao final, imprima na tela a quantidade de cores encontradas (na primeira linha) e qual é a cor correspondente a cada vértice (segunda linha), como no exemplo de 6 vértices na saída abaixo (vértice 1 tem a cor 2, vértice 2 tem a cor 1, vértice 3 tem a cor 2, vértice 4 tem a cor 1, vértice 5 tem a cor 3 e vértice 6 tem a cor 2).

3
2, 1, 2, 1, 3, 2

4. **[Relatório]** (2,5pts) Elabore um relatório em PDF de uma página comentando para cada um dos exercícios quais as estruturas de dados selecionadas, justificando as escolhas. Não esqueça de informar os nomes dos integrantes da equipe.

Para o caso de grafos dirigidos, a palavra “*arcs” aparece no lugar de “*edges”