

**Teoria dos Grafos**  
**COS 242**  
2012/2

**Trabalho de Disciplina – Parte 1**  
**Entrega e Apresentação - 21/11/2012**

## 1 Objetivos

O objetivo deste trabalho de disciplina é projetar e desenvolver uma biblioteca para manipular grafos. A biblioteca deverá ser capaz de representar grafos assim como implementar um conjunto de algoritmos em grafos. Para fins deste trabalho, a biblioteca pode ser um conjunto de funções (em C, por exemplo) ou uma classe em uma linguagem orientada a objetos (em C++ ou Java por exemplo). Você deve projetar e desenvolver sua biblioteca de forma que ela possa ser facilmente utilizada em outros programas.

## 2 Logística

Este trabalho de disciplina está divido em três partes. Cada parte corresponde a uma ou mais funcionalidades que deverão ser projetadas, desenvolvidas e incorporadas à sua biblioteca. O trabalho deverá ser realizado em dupla. Cada parte terá um prazo de entrega e será avaliada de maneira independente. Para entregar uma parte do trabalho, você irá preparar um relatório informando suas decisões de projeto e implementação das funcionalidades. Além disso, seu relatório deve responder as perguntas relacionadas aos estudos de caso. Este relatório deve ter no **máximo 4 páginas** e será entregue juntamente com o código fonte da biblioteca e do programa utilizado para realizar os estudos de caso. Além disso, cada parte do trabalho será apresentada em aula pela dupla. A apresentação deve ter em torno de 10 minutos e deve explicar as decisões de projeto e implementação que foram tomadas.

## 3 Descrição – Parte 1

Segue abaixo as funcionalidades que precisam ser oferecidas pela biblioteca na primeira parte do trabalho. Sua biblioteca irá trabalhar apenas com grafos não-direcionados.



Figura 1: Exemplo de grafo e formato de arquivos.

1. **Entrada.** Sua biblioteca deve ser capaz de ler um grafo de um arquivo texto. O formato do grafo no arquivo será o seguinte. A primeira linha informa o número de vértices do grafo. Cada linha subsequente informa as arestas. Um exemplo de um grafo e seu respectivo arquivo texto é dado na figura 1.

2. **Saída.** Sua biblioteca deve ser capaz de gerar um arquivo texto com as seguintes informações sobre o grafo: número de vértices, número de arestas, grau médio, e distribuição empírica do grau dos vértices<sup>1</sup>. A Figura 1 ilustra o formato deste arquivo de saída para o grafo correspondente.
3. **Representação de grafos.** Sua biblioteca deve ser capaz de representar grafos utilizando tanto uma matriz de adjacência, quanto uma lista de adjacência. O usuário da biblioteca (programa que irá usá-la) poderá escolher a representação a ser utilizada.
4. **Busca em grafos: largura e profundidade.** Sua biblioteca deve ser capaz de percorrer o grafo utilizando busca em largura e busca em profundidade. O vértice inicial será dado pelo usuário da biblioteca. A respectiva árvore de busca deve ser gerada assim como o nível de cada vértice na árvore (nível da raiz é zero). Estas informações devem ser impressas em um arquivo. Para descrever a árvore gerada, basta informar o pai de cada vértice e seu nível no arquivo de saída.
5. **Componentes conexos.** Sua biblioteca deve ser capaz descobrir os componentes conexos de um grafo. O número de componentes conexas, assim como o tamanho (em vértices) de cada componente e a lista de vértices pertencentes à componente. Os componentes devem estar listados em ordem decrescente de tamanho (listar primeiro o componente com o maior número de vértices, etc).

## 4 Estudo de Caso 1

Considere o grafo de colaboração entre pesquisadores disponível no website da disciplina. Utilizando este grafo, responda às perguntas abaixo.

1. Compare o desempenho em termos de quantidade de memória utilizada das duas representações do grafo. Determine a quantidade de memória (em MB) utilizada pelo seu programa quando você representa o grafo utilizando uma matriz de adjacência e lista de adjacência.
2. Compare o desempenho em termos de tempo de execução das duas representações do grafo. Determine o tempo necessário para executar uma busca em largura em cada um dos casos (utilize diferentes vértices como ponto de partida da busca).
3. Obtenha a distribuição empírica do grau dos vértices. Trace um gráfico com seu resultado. Qual é o maior grau do grafo? E o menor? Como isto se compara ao maior grau possível?
4. Obtenha os componentes conexos do grafo. Quantos componentes conexos tem o grafo? Qual é o tamanho do maior e do menor componente conexo?

## 5 Estudo de Caso 2

Considere o grafo de conexão das redes que formam a Internet (AS Graph). Parte deste grafo estará disponível em breve no formato descrito neste trabalho (no website). Utilizando este grafo, responda às perguntas abaixo.

1. Obtenha a distribuição empírica do grau dos vértices. Trace um gráfico com seu resultado. Qual é o maior grau do grafo? E o menor? Como isto se compara ao maior grau possível?

---

<sup>1</sup>A distribuição empírica do grau dos vértices é dada pela frequência relativa dos graus. Ou seja,  $f(d) = n(d)/n$ , onde  $f(d)$  é a frequência relativa do grau  $d$ , que é dada pelo número de vértices com grau  $d$  (dado por  $n(d)$ ) dividido pelo número total de vértices do grafo (dado por  $n$ ).

2. Obtenha os componentes conexos do grafo. Quantos componentes conexos tem o grafo? Qual é o tamanho do maior e do menor componente conexo?
3. Faça uma busca em largura à partir do vértice 1. Neste caso, o maior nível da árvore geradora de busca representa a maior distância do vértice 1 a qualquer outro. Determine este valor. Repita este procedimento para outros vértices. O que você pode concluir?
4. Determine o diâmetro da Internet. Lembrando que o diâmetro é a maior distância entre qualquer par de vértices do grafo (ou seja, o comprimento do maior caminho mínimo do grafo). Utilize a BFS para responder esta pergunta.