



Lista 2

Entrega: até 23h55 do dia 28/10/2019

- Submeta ao tidia um único arquivo **.pdf** com as suas soluções escaneadas dos exercícios teóricos (sugestão de aplicativo: CamScanner) e um único arquivo **.c** com as soluções dos exercícios práticos.
- Seja o mais **formal** possível em todas as respostas.
- Identifique devidamente cada exercício.
- Capriche na letra!
- A lista é uma forma de treino para a prova, que não terá consulta. Evite plágio!

1 Exercícios teóricos

1. (1 PONTO) Todo grafo euleriano bipartido possui um número par de arestas?
2. (1.5 PONTO) Se G é um grafo conexo com $2k > 0$ vértices de grau ímpar, então G pode ser decomposto em k trilhas.
3. (1.5 PONTO) Prove que toda árvore tem somente uma bipartição.
4. (1 PONTO) Prove que toda árvore tem uma folha no maior conjunto da partição.
5. (1 PONTO) Mostre que o seguinte vale para todo grafo G com n vértices e m arestas: $m \geq n - c(G)$, onde $c(G)$ denota a quantidade de componentes conexas de G . Mostre que a igualdade vale apenas quando G é uma floresta.
6. (1 PONTO) Prove que toda árvore com grau máximo $\Delta(G) \geq 2$ tem pelo menos $\Delta(G)$ folhas. Mostre que isso é o melhor possível construindo uma árvore de ordem n com exatamente $\Delta(G)$ folhas, para cada escolha de n e $\Delta(G)$, com $n > \Delta(G) \geq 2$.
7. (1 PONTO) Considere o grafo $G = (V, E)$ definido por $V(G) = \{a, b, c, d, e, f, g, h, i\}$ e $E(G) = \{ad, de, ea, ba, bf, fg, gb, cb, ch, hi, ic\}$. Execute a busca em largura sobre G a partir do vértice e e indique a distância de cada vértice até e .

2 Exercícios práticos

1. (2 PONTOS) Implemente um arquivo `grafo.c` cujo arquivo cabeçalho¹, `grafo.h`, é o seguinte:

```
1  #ifndef __GRAFO_H
2  #define __GRAFO_H
3
4  typedef struct grafo grafo_t;
5  grafo_t* cria_grafo(int n, int m);
6  void adiciona_aresta(grafo_t *G, int u, int v);
7  void imprime_grafo(grafo_t *G);
8  void deleta_grafo(grafo_t *G);
9  int grau(grafo_t *G, int v);
10 int grau_maximo(grafo_t *G);
11 int* DFS(grafo_t *G, int s);
12 int eh_aresta_corte(grafo_t *G, int u, int v);
13 int eh_par(grafo_t *G);
14 int* Fleury(grafo_t *G, int v);
15
16 #endif
```

O arquivo `grafo.h` completo com explicações das funções (não modifique-o!) e outros arquivos auxiliares encontram-se em <http://professor.ufabc.edu.br/~carla.negri/cursos/materiais/implementacoes/>.

3 Exercícios extras

1. Seja G um grafo conexo onde todos os vértices têm grau par. Prove que, para todo vértice v de $V(G)$, a quantidade de componentes de $G - v$ é no máximo $d(v)/2$.
2. Se G é um grafo conexo e tem exatamente um ciclo, qual a quantidade de árvores geradoras de G ? E se G é conexo e tem exatamente dois ciclos?
3. Prove que um grafo G é uma floresta se e somente se todo subgrafo conexo de G é um subgrafo induzido.
4. Explique como usar a BFS para calcular a cintura de um grafo.
5. Prove que o algoritmo a seguir encontra corretamente o diâmetro de uma árvore. Execute a BFS a partir de um vértice w qualquer para encontrar um vértice u que está à maior distância de w . Execute a BFS novamente, agora a partir de u , para encontrar um vértice v que está à maior distância de u . Retorne $dist(u, v)$.
6. Escreva um algoritmo para decidir se um grafo é bipartido.
7. Existe grafo euleriano simples com número par de vértices e ímpar de arestas?
8. Sejam d_1, d_2, \dots, d_n inteiros positivos com $n \geq 2$. Prove que existe uma árvore cujos vértices têm graus d_1, d_2, \dots, d_n se e somente se $\sum_{i=1}^n d_i = 2n - 2$.

¹Você pode ler um pouco sobre arquivos cabeçalho aqui: http://www.ppgia.pucpr.br/~laplima/ensino/tap/contents/02_arquivosh.html