Teoria dos Grafos 2020/1S
Exercícios 2

1) Forneça um algoritmo com complexidade de tempo $O(V)$ que determine se um dado grafo não direcionado $G=(V,E)$ contém um ciclo.

2) Seja *T* uma árvore de busca em largura de um grafo conexo *G*, com raiz *r*. Mostre que para cada vértice *v* de *G*, $l(v) = d_T(r,v)$.

3) Considere o jogo em que o jogador recebe um grafo conexo. O objetivo é remover os vértices deste grafo, um a um, sem desconectar o grafo em nenhum momento. O jogador vence se conseguir remover todos os vértices.
a) Sempre é possível vencer o jogo? Justifique.
b) Descreva um algoritmo para vencer o jogo nos casos em que é possível vencer.

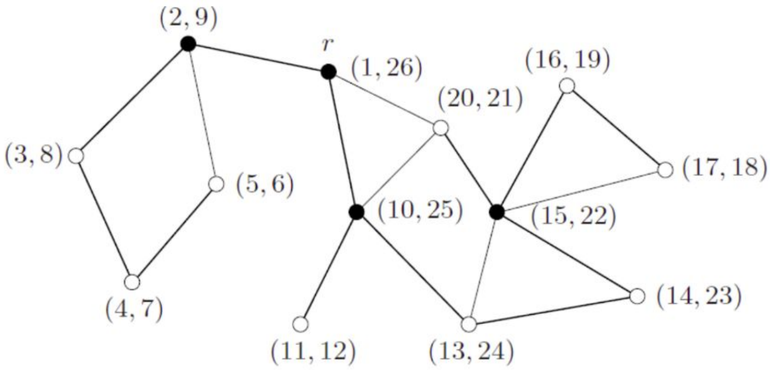
4) Mostre que todo grafo conexo contém um vértice que não é vértice de corte.

5) Prove ou mostre um contra-exemplo para as afirmações abaixo.
   a) Todo grafo sem vértice de corte não tem aresta de corte.
   b) Todo grafo sem aresta de corte não tem vértice de corte.

6) Seja G um grafo conexo e T uma árvore de busca em profundidade de G, em que todas as arestas são orientadas do pai para o filho e as arestas de volta (backedge) são orientadas do descendente para um ancestral.
   a) Mostre que (ver definição de f* nos slides sobre busca - vértices de corte)

$$f^*(v) = \min\{f(v), \quad \min_{backedge(v,w)} f(w), \quad \min_{u \in filho(v)} f^*(u)\}$$

   b) Seja o grafo *G* abaixo com os valores de *f(v)* e *l(v)* para cada vértice *v* de *G*. Calcule *f*(v)* para cada vértice *v* de *G*.



   c) Forneça uma adaptação do algoritmo DFS que calcule os valores $f^*(v)$. Qual a sua complexidade do tempo de execução?