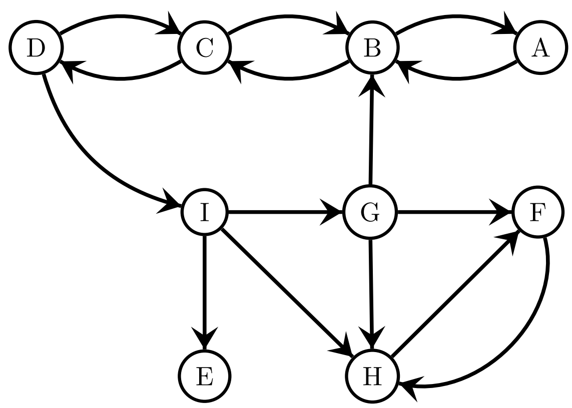
Aula Prática 3

ASA 2020/2021

R1 16/17 I.b Considere o grafo dirigido:



Aplique o algoritmo de Tarjan no grafo. Considere que inicia a sua travessia pelo vértice D e que os vértices são explorados por ordem alfabética. Indique os vértices de cada componente fortemente ligado do grafo. Indique os tempos de descoberta d e de *low* de cada vértice.

Nota: Neste algoritmo os valores de d começam em 1.

Ex 22.5-3 O professor Bacon afirma que o algoritmo para componentes fortemente ligados seria mais simples se usássemos o grafo original (em vez do transposto) na segunda DFS e, adicionalmente, analisássemos os nós por ordem crescente dos seus tempos de fim.

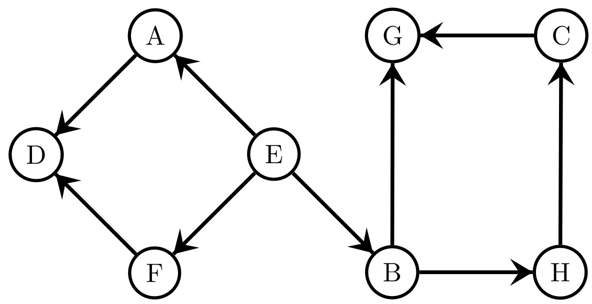
Será que o algoritmo produz sempre os resultados correctos?

Ex 22.5-1 Como muda o número de componentes fortemente ligados de um grafo se adicionarmos um novo caminho entre nós?

Ex 22.4-3 Indique um algoritmo que determine se um grafo não dirigido $G = (V, E)$ contém um ciclo simples. O algoritmo deve correr em tempo $O(V)$ independentemente de E.

Ex 22.5-5 Indique um algoritmo em tempo $O(V + E)$ para calcular o grafo de componentes fortemente ligado dado um grafo dirigido $G = (V, E)$. Garanta que existe no máximo um arco entre dois vértices do grafo de componentes produzido.

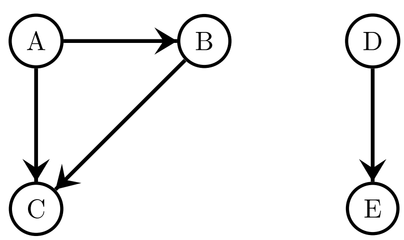
T1 12/13 – I.b Considere o grafo dirigido acíclico DAG:



Aplique uma procura em profundidade primeiro (DFS) com início no vértice E e que visita os filhos por ordem lexicográfica. Indique os valores de descoberta (d) e fim (f), para cada um dos vértices. Indique também a ordenação topológica resultante.

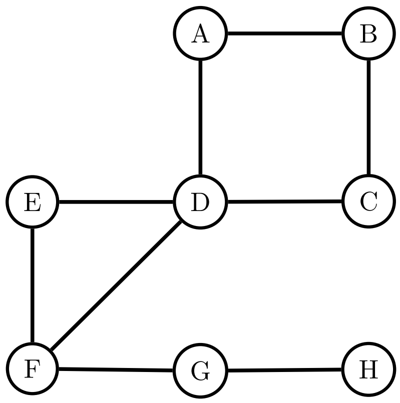
Nota: Os tempos de uma DFS começam em 1.

T1 06/07 I.2 Considere o seguinte grafo dirigido:



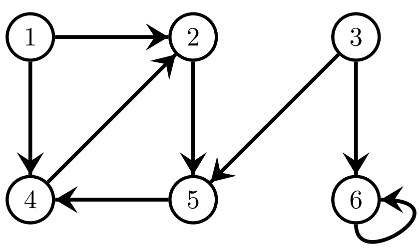
Indique três ordenações topológicas possíveis para este grafo. Indique os valores de descoberta e fim (d, f) que levaram a essas ordenações.

T1 06/07 I.1 Considere o grafo não dirigido seguinte.



Para todos os nós u, calcule $d[u]$ e $\pi[u]$ obtidos por uma procura em largura primeiro a partir do nó A.

Ex. 22.2-1 Aplique uma BFS no seguinte grafo, a começar em 3 e utilizando a ordem númerica para os vizinhos. Indique também o d e π de cada valor.



Ex. 22.2-7 Proponha um algoritmo $O(V + E)$ que reconheça se um dado grafo é bipartido. Produza a partição ou mostre que tal partição não existe.

R1 08/09 I.3 Considere a aplicação de uma pesquisa em largura (BFS) num grafo $G = (V, E)$, onde $s \in V$ é o vértice origem da BFS. Considere ainda a utilização da árvore BF para classificação dos arcos, tal como na execução de uma pesquisa em profundidade (DFS). Assim, para cada uma das seguintes afirmações, indique se é verdadeira (V) ou falsa (F).

1. A BFS permite identificar os caminhos mais curtos para todos os vértices do grafo atingíveis a partir de s.
2. Sejam u e v vértices do grafo atingíveis a partir de s tal que $d[v] > d[u]+1$. Nesse caso, o arco (u, v) não existe no grafo.
3. Se o grafo G for não dirigido, na aplicação de uma BFS podem existir arcos para a frente.
4. Sejam u e v dois vértices do grafo atingíveis a partir de s tal que $d[v] > d[u]$. Então temos necessariamente que $d[v] - d[u]$ denota o número de arcos no caminho mais curto de u para v.
5. Para cada arco (u, v) da árvore BF temos que $d[v] = d[u] + 1$.
6. Se o grafo G for não dirigido, na aplicação de uma BFS não existem arcos de cruzamento.