Projeto e Análise de Algoritmos

Flávio L. C. de Moura*

29 de março de 2022

1 Busca em Profundidade

Nesta seção estamos interessados em resolver o seguinte problema: dados dois vértices u e v de um (di)grafo G, existe um caminho de u para v? Este problema é conhecido como o *problema da acessibilidade*. O algoritmo DFS a seguir resolve este problema:

Algorithm 1: DFS(G)

```
1  for each vertex u ∈ G.V do
2      u.color = WHITE;
3      u.π = NIL;
4  end
5  time = 0;
6  for each vertex u ∈ G.V do
7      if u.color == WHITE then
8          DFS-Visit(G, u)
9      end
10 end
```

Algorithm 2: DFS-Visit(G, u)

```
1  time = time + 1;
2  u.d = time;
3  u.color = GRAY;
4  for each v ∈ G.Adj[u] do
5      if v.color == WHITE then
6          v.π = u;
7          DFS-Visit(G, v)
8      end
9  end
10 u.color = BLACK;
11 time = time + 1;
12 u.f = time;
```

Qual o tempo de execução de DFS? Os *loops* das linhas 1-4 e 6-10 são executados em tempo $\Theta(V)$. O procedimento DFS-Visit é chamado exatamente uma vez para cada vértice v de G. Em cada execução de DFS-Visit(G, v), o *loop* das linhas 4-9 é executado $|Adj[v]|$ vezes. Como

$$\sum_{v \in V} |Adj[v]| = \Theta(E)$$

o custo total de DFS-Visit é $\Theta(E)$, e portanto, o custo total de DFS é $\Theta(V + E)$.

Teorema 1.1. *O problema da acessibilidade em um (di)grafo $G = (V, E)$ pode ser resolvido em tempo $\Theta(V + E)$.*

*flavio@flaviomoura.info

Definição 1.2. *O* subgrafo predecessor *da busca em profundidade é definido por:*

- $G_\pi = (V, E_\pi)$, *onde* $E_\pi = \{(v.\pi, v) : v \in V \text{ and } v.\pi \neq NIL\}$

O subgrafo predecessor de uma busca em profundidade forma uma floresta.

Teorema 1.3. *Para dois vértices u e v quaisquer de um (di)grafo G, apenas uma das seguintes propriedades ocorre em uma busca em profundidade (DFS) em G, considerando que* $u.d < v.d$*:*

1. $u.d < v.d < v.f < u.f$, *e v é um descendente de u no subgrafo predecessor de G;*

2. $u.d < u.f < v.d < v.f$, *e u não é um descendente de v no subgrafo predecessor de G, ou vice-versa.*

Teorema 1.4. *Seja* $G = (V, E)$ *um grafo, e considere a floresta F obtida após a execução de DFS(G). As componentes de F são precisamente as componentes conexas de G.*

Corolário 1.5. *As componentes conexas de um grafo* $G = (V, E)$ *podem ser encontradas em tempo* $\Theta(V + E)$.