

Depth-first search

La ricerca in profondità visita i nodi di un grafo esplorandoli a partire dall'ultimo nodo che è stato scoperto. Una volta terminata la visita di una componente connessa, si sceglie un nodo non ancora esplorato ripetendo la ricerca finché tutti i nodi non sono stati raggiunti. Il sottografo dei predecessori è una *foresta depth-first*.

Per ogni nodo v DFS registra il predecessore, colore, tempo di scoperta e tempo di fine dell'esplorazione dei nodi adiacenti. I timestamp sono interi tra 1 e $2|V|$, dal momento che c'è un evento di scoperta e uno di fine per ogni nodo. v è bianco prima di $v.d$, grigio tra $v.d$ e $v.f$, nero dopo $v.f$.

DFS(G)

```
1  for  $u \in G.V$ 
2       $u.color = \text{WHITE}$ 
3       $u.pred = \text{NIL}$ 
4  global  $time = 0$ 
5
6  for  $u \in G.V$ 
7      if  $u.color == \text{WHITE}$ 
8          DFS-VISIT( $G, u$ )
```

DFS-VISIT(G, u)

```
1   $time = time + 1$ 
2   $u.d = time$ 
3   $u.color = \text{GRAY}$ 
4  for  $v \in G.adj[u]$ 
5      if  $v.color == \text{WHITE}$ 
6           $v.pred = u$ 
7          DFS-VISIT( $G, v$ )
8   $u.color = \text{BLACK}$ 
9   $time = time + 1$ 
10  $u.f = time$ 
```

Complessità

DFS-VISIT è chiamata esattamente una volta per ogni nodo v in V , dal momento che viene chiamata solo su nodi bianchi e colora immediatamente di grigio il suo input. Il ciclo al suo interno viene eseguito $|G.adj[v]|$ volte, quindi in totale $\Theta(E)$. DFS ha costo $\Theta(V)$ escluso DFS-VISIT, quindi in totale $\Theta(V + E)$.