

Typografie a publikování – 5. projekt

Breadth-First Search Algorithm

Denys Petrovskyi

8. května 2023

- Co je BFS a jak se používá?
- Jaké jsou výhody BFS?
- Příklady

Definice BFS

- BFS je algoritmus procházení, který začíná v kořenovém uzlu a zkoumá všechny uzly v aktuální hloubce, než se přesune k uzlům v další hloubce.
- BFS navštíví všechny uzly v grafu, ale pořadí, ve kterém jsou uzly navštíveny, závisí na struktuře grafu.
- BFS lze použít k nalezení nejkratší cesty mezi dvěma uzly v neváženém grafu.

Složitost algoritmu

Časovou složitost lze vyjádřit jako $O(|V| + |E|)$, protože v nejhorším případě bude prozkoumán každý vrchol a každá hrana.

- $|V|$ je počet vrcholů v grafu
- $|E|$ je počet hran v grafu

Prostorová složitost je $O(B^M)$, kde:

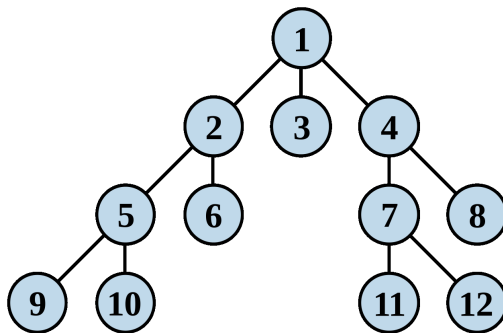
- B je nejvyšší stupeň větvení stromu
- M je nejvyšší délka cesty ve stromě

Pseudokód

Input : Graf G a počáteční uzel s

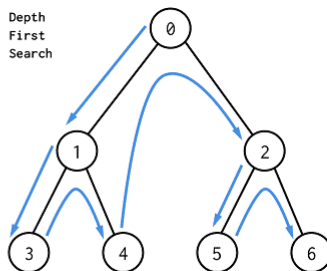
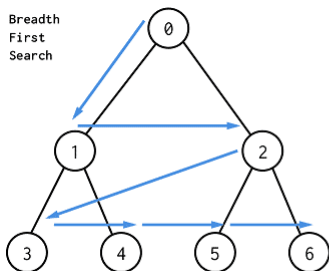
Output: Nejkratší cesta z s do všech ostatních uzlů v G

```
for  $u \in U(G) - s$  do  
    |  $stav[u] = FRESH$ ;  $d[u] = \infty$ ;  $p[u] = null$ ;  
end  
 $stav[s] = OPEN$ ;  $d[s] = 0$ ;  $p[s] = null$ ;  $Queue.Init()$ ;  $Queue.Push(s)$ ;  
while  $!Queue.Empty()$  do  
    |  $u = Queue.Pop()$ ;  
    | for  $v \in Adj[u]$  do  
        | if  $stav[v] == FRESH$  then  
            |  $stav[v] = OPEN$ ;  $d[v] = d[u] + 1$ ;  $p[v] = u$ ;  $Queue.Push(v)$ ;  
        | end  
    | end  
    |  $stav[u] = CLOSED$ ;  
end
```



Obrázek: Příklad BFS

Obrázky



Obrázek: Srovnání BFS a DFS

Závěr

- BFS je základní grafový algoritmus, který se používá v mnoha aplikacích
- Má časovou složitost $O(|V| + |E|)$ a prostorovou složitost $O(B^M)$
- BFS lze použít k nalezení nejkratší cesty mezi dvěma vrcholy, kontrole, zda je graf propojen, a zjištění průměru grafu