CS3000: Algorithms & Data — Summer '25 — Laney Strange

Breadth-First Search

The algorithm below works on a graph represented in a adjacency list. We have an array Adj of $|V|$ lists, one for each vertex in V. For each $u \in V$, the adjacency list $Adj[u]$ contains all the vertices v such that there is an edge $(u, v) \in E$.

Breadth-First Search (BFS) is a simple algorithm whose job it is to explore a graph. Starting from a given source source vertex s, it systematically explores the edges of the graph G to discover every vertex that is reachable from s. It uses a queue, Q, to manage the vertices in the middle of being explored.

BFS(G, s)
1 **for** each vertex $u \in G.V - \{s\}$
2 $\quad u.color = \textsc{White}$
3 $\quad u.d = \infty$
4 $\quad u.\pi = \textsc{Nil}$
5 $s.color = \textsc{Gray}$
6 $s.d = 0$
7 $s.\pi = \textsc{Nil}$
8 $Q = \emptyset$
9 $\textsc{Enqueue}(Q, s)$
10 **while** $Q \neq \emptyset$
11 $\quad u = \textsc{Dequeue}(Q)$
12 $\quad$ **for** each $v \in G.Adj[u]$
13 $\quad\quad$ **if** $v.color == \textsc{White}$
14 $\quad\quad\quad v.color = \textsc{Gray}$
15 $\quad\quad\quad v.d = u.d + 1$
16 $\quad\quad\quad v.\pi = u$
17 $\quad\quad\quad \textsc{Enqueue}(Q, v)$
18 $\quad u.color = \textsc{Black}$

We typeset the BFS procedure above with the following LaTeX:

```
\begin{codebox}
\Procname{$\proc{BFS}(G, s)$}
\li \For each vertex in $u \in G.V - \{s\}$
\Do
\li $\id{u.color} \gets \const{White}$
\li $u.d \gets \infty$
\li $u.\pi \gets \const{Nil}$
\End
\li $\id{s.color} \gets \const{Gray}$
\li $s.d \gets 0$
```

```
\li $s.\pi \gets \const{Nil}$
\li $Q \gets \emptyset$
\li $\proc{Enqueue}(Q, s)$
\li \While $Q \ne \emptyset$
\Do
\li $u \gets \proc{Dequeue}(Q)$
\li \For each $v \in G.Adj[u]$
\Do
\li \If $\id{v.color} == \const{White}$
\Then
\li $\id{v.color} \gets \const{Gray}$
\li $v.d \gets u.d + 1$
\li $v.\pi \gets u$
\li $\proc{Enqueue}(Q, v)$
\End
\End
\li $\id{u.color} \gets \const{Black}$
\end{codebox}
```