Algorithm Lab

Supta Philip Philip

November 1, 2023

1 Graph Algorithm

Implement the following graph algorithms using C++.

1. BFS

2. DFS Recursive

3. DFS Iterative

Algorithm 1 BFS

1: **procedure** BFS(G, v)
2:      $queue < int > Q$
3:      $Q.push(v)$
4:      $visited[v] = TRUE$
5:      $distance[v] = 0$
6:      **while** $!Q.empty()$ **do**
7:          $p = Q.front()$
8:          $Q.pop()$
9:          **for** $i \leftarrow 0, G[v].size()$ **do**
10:             $next = graph[v][i]$
11:             **if** $visited[next] == 0$ **then**
12:                 $Q.push(next)$
13:                 $visited[next] = 1$
14:                 $distance[next] = distance[v] + 1$
15:             **end if**
16:          **end for**
17:      **end while**
18: **end procedure**

Algorithm 2 DFS Recursive

1: **procedure** DFS(G, v)
2:     $visited[v] = TRUE$
3:     **for** $i \leftarrow 0, G[v].size()$ **do**
4:         $next = graph[v][i]$
5:         **if** $visited[next] == 0$ **then**
6:             $DFS(next)$
7:         **end if**
8:     **end for**
9: **end procedure**

Algorithm 3 DFS Iterative

1: **procedure** DFS(G, v)
2:     $Stack < int > S$
3:     $S.push(v)$
4:     $visited[v] = TRUE$
5:     **while** $!S.empty()$ **do**
6:         $p = S.top()$
7:         $S.pop()$
8:         **for** $i \leftarrow 0, G[v].size()$ **do**
9:             $next = graph[v][i]$
10:             **if** $visited[next] == 0$ **then**
11:                 $S.push(next)$
12:                 $visited[next] = 1$
13:             **end if**
14:         **end for**
15:     **end while**
16: **end procedure**