```
BFS(V, E, s)

1.         for each u in V − {s}             ▷ for each vertex u in V[G] except s.
2.             do color[u] ← WHITE
3.                 d[u] ← infinity
4.                 π[u] ← NIL
5.         color[s] ← GRAY                ▷ Source vertex discovered
6.         d[s] ← 0                             ▷ initialize
7.         π[s] ← NIL                        ▷ initialize
8.         Q ← {}                             ▷ Clear queue Q
9.         ENQUEUE(Q, s)
10        while Q is non-empty
11.             do u ← DEQUEUE(Q)                        ▷ That is, u = head[Q]
12.                 for each v adjacent to u                  ▷ for loop for every
node along with edge.
13.                         do if color[v] ← WHITE        ▷ if color is white you've
never seen it before
14.                                    then  color[v] ← GRAY
15.                                          d[v] ← d[u] + 1
16.                                          π[v] ← u
17.                                          ENQUEUE(Q, v)
18.                 DEQUEUE(Q)
19.          color[u] ← BLACK


DFS (V, E)

1.     for each vertex u in V[G]
2.        do color[u] ← WHITE
3.                π[u] ← NIL
4.     time ← 0
5.     for each vertex u in V[G]
6.        do if color[u] ← WHITE
7.                then DFS-Visit(u)              ▷ build a new DFS-tree from u



DFS-Visit(u)

1.     color[u] ← GRAY                          ▷ discover u
2.     time ← time + 1
3.     d[u] ← time
4.     for each vertex v adjacent to u     ▷ explore (u, v)
5.        do if color[v] ← WHITE
6.                then π[v] ← u
7.                     DFS-Visit(v)
8.     color[u] ← BLACK
9.     time ← time + 1
10.   f[u] ← time                                ▷ we are done with u
```