



Graph Traversals

Learning Objectives

1. Implement DFS for Graph Traversals



Graph Traversal - BFS

```
1 BFS(G) :  
2   Input: Graph, G  
3   Output: A labeling of the edges on  
4           G as discovery and cross edges  
5  
6   foreach (Vertex v : G.vertices()):  
7       setLabel(v, UNEXPLORED)  
8   foreach (Edge e : G.edges()):  
9       setLabel(e, UNEXPLORED)  
10  foreach (Vertex v : G.vertices()):  
11      if getLabel(v) == UNEXPLORED:  
12          BFS(G, v)
```

Discovery Edge

- An edge taken to find a new node

Cross Edge

- An edge taken to find an already visited node



Graph Traversal - DFS

```
1 DFS (G) :  
2   Input: Graph, G  
3   Output: A labeling of the edges on  
4           G as discovery and back edges  
5  
6   foreach (Vertex v : G.vertices()):  
7       setLabel(v, UNEXPLORED)  
8   foreach (Edge e : G.edges()):  
9       setLabel(e, UNEXPLORED)  
10  foreach (Vertex v : G.vertices()):  
11      if getLabel(v) == UNEXPLORED:  
12          DFS (G, v)
```

Discovery Edge

- An edge taken to find a new node

Back Edge

- An edge taken to find an already visited node

Graph Traversal - BFS from a vertex

```
14 BFS(G, v):
15     Queue q
16     q.enqueue(v)
17
18     while !q.empty():
19         v = q.dequeue()
20         If (getLabel(v) != VISITED){
21             setLabel(v, VISITED)
22             foreach (Vertex w : G.adjacent(v)):
23                 if getLabel(w) == UNEXPLORED:
24                     setLabel(v, w, DISCOVERY)
25                     setLabel(w, VISITED)
26                     q.enqueue(w)
27                 elseif getLabel(v, w) == UNEXPLORED:
28                     setLabel(v, w, CROSS)
29         }
```

Node Labels

- Visited
- Unexplored

Edge Labels

- Discovery
- Cross
- Unexplored

Graph Traversal - DFS from a vertex

```
14 DFS(G, v):
15     Stack s
16     s.enqueue(v, NULL)
17
18     while !s.empty():
19         [v, e] = s.pop()
20         if (getLabel(v) == UNEXPLORED){
21             setLabel(v, VISITED)
22             if (e != NULL) {
23                 setLabel(e, DISCOVERY)
24                 computeInfo(e, v)
25             }
26             foreach (Vertex w : G.adjacent(v)):
27                 s.push(w, Edge(v, w))
28         elseif getLabel(e) == UNEXPLORED:
29             setLabel(e, Back)
30     }
```

Node Labels

- Visited
- Unexplored

Edge Labels

- Discovery
- Back
- Unexplored

Graph Traversal - DFS Recursive

```
14 DFS(G, v):
15     setLabel(v, VISITED)
16     for each w in G.adjacent(v):
17         e = Edge(v, w)
18         if getLabel(w) == UNEXPLORED:
19             setLabel(e, DISCOVERY)
20             DFS(G, w)
21         elif getLabel(e) == UNEXPLORED:
22             setLabel(e, BACK)
23
```

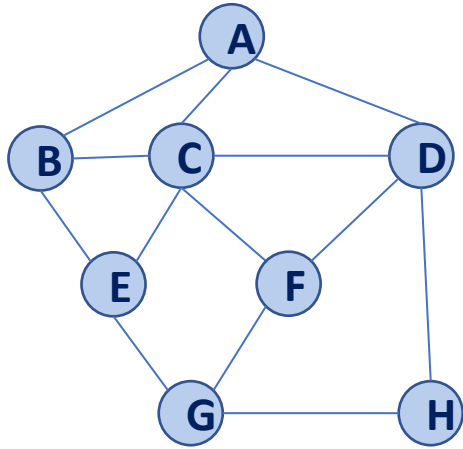
Node Labels

- Visited
- Unexplored

Edge Labels

- Discovery
- Back
- Unexplored

DFS Traversal



d - distance

P - predecessor

Stack

v	d	P	Adjacent Edges
A	0	-	C B D
B			A C E
C			B A D E F
D			A C F H
E			B C G
F			C D G
G			E F H
H			D G