

DFS ALGORITHM

```
// DFS algorithm in Java
import java.util.*;

class Graph {
    private LinkedList<Integer> adjLists[];
    private boolean visited[];

    // Graph creation
    Graph(int vertices) {
        adjLists = new LinkedList[vertices];
        visited = new boolean[vertices];
        for (int i = 0; i < vertices; i++)
            adjLists[i] = new LinkedList<Integer>();
    }

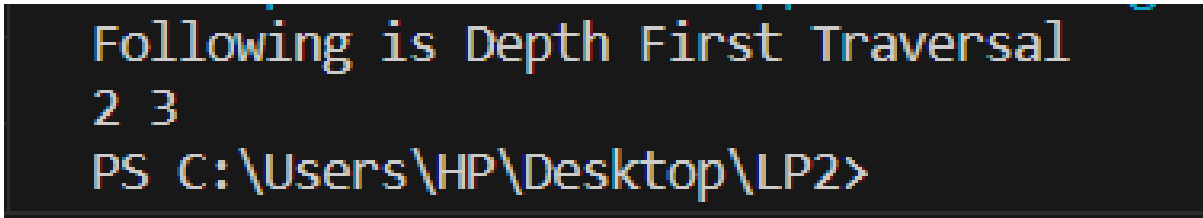
    // Add edges
    void addEdge(int src, int dest) {
        adjLists[src].add(dest);
    }

    // DFS algorithm
    void DFS(int vertex) {
        visited[vertex] = true;
        System.out.print(vertex + " ");
        Iterator<Integer> ite = adjLists[vertex].listIterator();
        while (ite.hasNext()) {
            int adj = ite.next();
```

```
if (!visited[adj])
    DFS(adj);
}
}

public static void main(String args[]) {
    Graph g = new Graph(4);
    g.addEdge(0, 1);
    g.addEdge(0, 2);
    g.addEdge(1, 2);
    g.addEdge(2, 3);
    System.out.println("Following is Depth First Traversal");
    g.DFS(2);
}
}
```

OUTPUT:



```
Following is Depth First Traversal
2 3
PS C:\Users\HP\Desktop\LP2>
```