

Breadth first search(BFS) 1

Problem

Submissions

Leaderboard

Discussions

Write a Java Program to print all nodes reachable from a given starting node in a digraph using BFS method

Solved: 106

Attempted: 107

Input Format

4 5 1 3 0 2 2 3 3 0 0 3 2

Constraints

positive vertex values only

Output Format

Enter the number of vertices : 4 Enter the number of edges : 5 Enter source : 1 Enter destination : 3 Enter source : 0 Enter destination : 2 Enter source : 2 Enter destination : 3 Enter source : 3 Enter destination : 0 Enter source : 0 Enter destination : 3 Enter Start Vertex for BFS : 2 BFS of graph : 2 3 0

Sample Input 0

4
5
1
3
0
2
2
3
3
0
0
3
2

Sample Output 0

BFS of graph : 2 3 0

[f](#) [t](#) [in](#)

Contest ends in 2 months

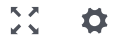
Submissions: 102

Max Score: 10

Difficulty: Medium

Rate This Challenge:

☆☆☆☆☆



```
1 //224G1A0553
2 import java.io.*;
3 import java.util.*;
4 import java.text.*;
5 import java.math.*;
6 import java.util.regex.*;
7 class Gnode {
8     Gnode next;
9     int vertex;
10 }
11 public class Solution {
12     Gnode graph[ ];
13     boolean visited[ ];
14     int queue[ ];
15     int numVertices;
16     int front;
17     int rear;
18     public Solution(int n) {
19         graph = new Gnode[n];
20         visited = new boolean[n];
21         queue = new int[n];
22         numVertices = n;
23         front = -1;
24         rear = -1;
25     }
26     void insertQueue(int vertex) {
27         if(rear == numVertices - 1){}
28         //System.out.printf("Queue Overflow.\n");
29         else {
30             if(front == -1)
31                 front = 0;
32             rear = rear + 1;
33             queue[rear] = vertex ;
34         }
35     }
36     boolean isEmptyQueue() {
37         if(front == -1 || front > rear)
38             return true;
39         else
40             return false;
41     }
42     int deleteQueue() {
43         int deleteItem;
44         if(isEmptyQueue()) {
45             //System.out.printf("Queue Underflow\n");
46             return -1;
47         }
48         deleteItem = queue[front];
49         front = front + 1;
50         return deleteItem;
51     }
52     void Bfs(int v) {
53         int w;
54         insertQueue(v);
55         while(!isEmptyQueue()) {
56             v = deleteQueue( );
57             System.out.printf(" %d",v);
58             visited[v] = true;
59             Gnode g = graph[v];
60             for( ; g != null; g = g.next) {
61                 w = g.vertex;
```

```
63         if(visited[w] == false) {
64             insertQueue(w);
65             visited[w] = true;
66         }}}}
67     public static void main(String []args) {
68         int n, e, i, s, d, v;
69         Gnode q, p;
70         Scanner sc = new Scanner(System.in);
71         //System.out.printf("Enter the number of vertices : ");
72         n = sc.nextInt();
73         //System.out.printf("Enter the number of edges : ");
74         e = sc.nextInt();
75         Solution g = new Solution(n);
76         for(i=1;i<=e;i++) {
77             //System.out.printf("Enter source : ");
78             s = sc.nextInt();
79             //System.out.printf("Enter destination : ");
80             d = sc.nextInt();
81             q = new Gnode();
82             q.vertex = d;
83             q.next = null;
84             if(g.graph[s] == null)
85                 g.graph[s]=q;
86             else {
87                 p=g.graph[s];
88                 while(p.next != null)
89                     p = p.next;
90                 p.next = q;
91             }
92             for(i = 0;i < n;i++)
93                 g.visited[i] = false;
94             //System.out.printf("Enter Start Vertex for BFS : ");
95             v = sc.nextInt();
96             System.out.printf("BFS of graph :");
97             g.Bfs(v);
98             System.out.printf("\n");
99         }}
```

Line: 35 Col: 11

[Upload Code as File](#) ☐ Test against custom input

Run Code

Submit Code

Testcase 0 **Congratulations, you passed the sample test case.**Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
4
5
1
3
0
2
2
3
3
0
0
3
2
```

Your Output (stdout)

```
BFS of graph : 2 3 0
```

Expected Output

```
BFS of graph : 2 3 0
```

[Interview Prep](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) |