

Implementation of BFS

```
1 import java.util.*;
2
3 class MyGraph {
4     Map<Integer, HashSet<Integer>> mp;
5     public MyGraph() {
6         mp = new HashMap<>();
7     }
8     public void addEdge(int v1, int v2, boolean isBiDir) {
9         HashSet<Integer> v1Neighbor = mp.getDefault(v1, new HashSet<>());
10        v1Neighbor.add(v2);
11        mp.put(v1, v1Neighbor);
12        if(isBiDir) addEdge(v2, v1, false);
13    }
14    public void display() {
15        for(Map.Entry<Integer, HashSet<Integer>> res : mp.entrySet()) {
16            System.out.println(res.getKey() + " -> " + res.getValue());
17        }
18    }
19    public void bfs(int src) {
20        Queue<Integer> BFS = new LinkedList<>();
21        BFS.add(src);
22        System.out.print(src);
23        HashSet<Integer> vis = new HashSet<>();
24        vis.add(src);
25        while(!BFS.isEmpty()) {
26            int temp = BFS.poll();
27            HashSet<Integer> Neighbours = mp.get(temp);
28            for (int tem : Neighbours) {
29                if (!vis.contains(tem)) {
30                    System.out.print("->" + tem);
31                    BFS.add(tem);
32                    vis.add(tem);
33                }
34            }
35        }
36        System.out.println();
37    }
38 }
39
40 public class Graph_with_Map {
41     public static void main(String[] args) {
42         MyGraph obj = new MyGraph();
43         obj.addEdge(1, 2, true);
44         obj.addEdge(1, 3, true);
45         obj.addEdge(3, 4, false);
46         obj.addEdge(3, 5, true);
47         obj.addEdge(5, 6, true);
48         obj.addEdge(2, 4, true);
49         obj.display();
50         obj.bfs(1);
51     }
52 }
```