Day 14

Assignment 5: Breadth-First Search (BFS) Implementation

For a given undirected graph, implement BFS to traverse the graph starting from a given node and print each node in the order it is visited.

A)

Let's implement Breadth-First Search (BFS) in Java to traverse an undirected graph. Here's a step-by-step guide:

- *1. Define Graph Class: We'll start by defining a Graph class to represent the graph using an adjacency list.*
- *2. Implement BFS Method: We'll implement a bfs method in the Graph class to perform BFS traversal starting from a given node.*
- *3.Create a Main Class: We'll create a separate class with a main method to demonstrate the usage of the BFS algorithm*

Java code :

```
package day14;
 import java.util.LinkedList;

 public class BfsGraph {

private int v;
private LinkedList<Integer>[] adjList;

public BfsGraph(int v) {
this.v = v;

adjList = new LinkedList[v];

for(int i=0;i<v;++i) {

adjList[i] = new LinkedList();
}
}

public void addEdge (int v, int w) {

adjList[v].add(w);

adjList[w].add(v);

}
```

```
public void BfsGraph(int s) {

boolean[] visited = new boolean[v];

LinkedList<Integer>queue = new LinkedList<Integer>();

visited[s] = true;

queue.add(s);

while(!queue.isEmpty()) {

int current = queue.poll();

System.out.print(current + ");
for(int neigbor : adjList[current]) {
if(!visited[neigbor]) {

visited[neigbor]= true;

queue.add(neigbor);

}

}

}


}

public static void main(String args[]) {

BfsGraph graph new BfsGraph (4);

for(int neigbor adjList[current]) {

graph.addEdge(0, 1); graph.addEdge(0, 2);

graph.addEdge(1, 2);

graph.addEdge(2, 0);

graph.addEdge(2, 3);

graph.addEdge(3, 3);
System.out.println("Bfs traversal starting from vertex 2:");

graph.BfsGraph (2);;

}

}
```

Explanation:

- *The Graph class represents an undirected graph using an adjacency list.*
- *The addEdge method is used to add an edge between two vertices in the graph.*
- *The bfs method performs BFS traversal starting from a given node. It uses a queue to keep track of nodes to visit next.*
- *The Main class demonstrates the usage of the Graph class by creating a graph, adding edges, and performing BFS traversal starting from node 0.*