

+ Ask New Question

More Related Questions

9 Public Followers

Last Asked Jul 5, 2016

16,017 Views

**Question Stats** 

Edits

Sponsored by Jira Software, Atlassian

Add a comment...

## Jira official site.

One tool to track issues & release great software. Try Jira for free.

Free trial at jira.com 000 Sravya Vissamsetty, God is REAL, unless explicitly declared INTEGER. (2)

Comment

Recommended All

BFS means a breadth first traversal. It means that you have to visit all the surrounding nodes of a particular node which are unvisited till then and then move on to the next node. continue the process till all the nodes are visited once.

## Algorithm:

- Rule 1 Visit adjacent unvisited vertex. Mark it visited. Display it. Insert it in a queue.
- Rule 2 If no adjacent vertex found, remove the first vertex from queue.
- Rule 3 Repeat Rule 1 and Rule 2 until queue is empty.

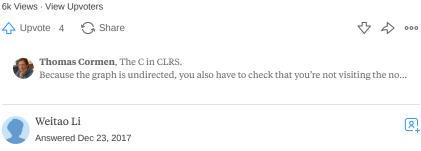
If not and no adjacent vertex present, add to code if(!visited(vertex)) bfs(vertex); something like this, so that for the next vertex the same procedure continues. for coding part of bfs go through this link: Breadth First Traversal for a Graph -GeeksforGeeks If the adjacent node of currect vertex has already been visited, then there is a loop hope this answers helps you. 6.6k Views · View Upvoters Your feedback is private. Is this answer still relevant and up to date? Yes No ♦ ♦ ••• ✓ Upvote · 3 Share Share Add a comment... Comment Recommended All Shivam Shivam, studied at DAV Public School (2015) (8) Answered Jun 8, 2017 Well use degree of each node .if degree of the node while adding to queue is greater than the current element whose neighbours are explored their ought to be cycle While(q is not empty) Current=deque(): For all neighboring element X of current If deg(X) >deg(current) Cycle detected 1.7k Views Upvote Share ? ♦ ♦ ••• Add a comment... Comment Recommended All Sponsored by Aha! What is a product roadmap? Build brilliant roadmaps in minutes. Trusted by over 200,000 users worldwide. Start a free 30-day trial. Free trial at aha.io 000

https://www.quora.com/How-can-I-detect-cycles-in-undirected-graph-using-BFS

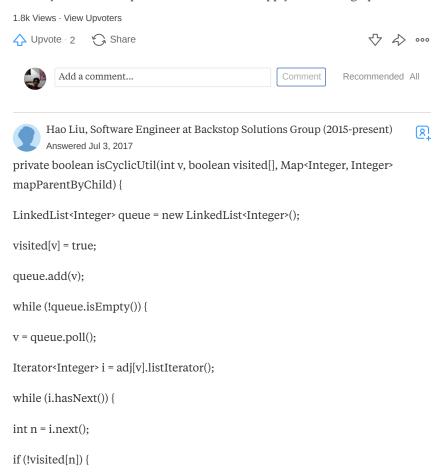
If you need to describe the cycle and not just detect its existence, you need to store the paths as you traverse the graph, which is why DFS might be more convenient.

Just perform BFS while keeping a list of previous nodes at each node visited, or else constructing a tree from the starting node. If I visit a node that has already been marked by BFS, I found a cycle; trace back along the path to describe it.

Remember that either will only work on the connected subgraph of the node you started at, not necessary find cycles on the whole graph.

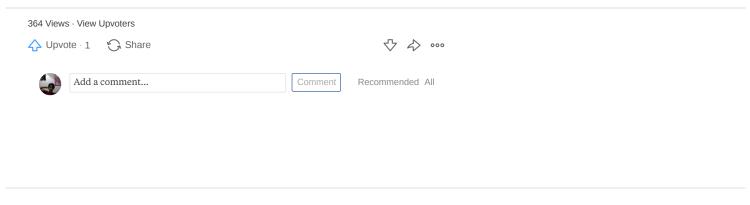


In bfs you have a visited list, so when you reading neighbors of current node and find there is a neighbor node which was visited before that means you found a loop. Think about this: the reason you reaching current node is because there is a path from root(first node you inserted into queue) to current node, and the reason this neighbor is visited is because there is a path from root to that node, now you found there is an edge between current node and neighbor which means you found a loop. However this does not apply to directed graph.



```
queue.offer(n);
mapParentByChild.put(n, v);
else {
Integer parent = mapParentByChild.get(v);
if (!parent.equals(n)) {
Set<Integer> setParents = new HashSet<Integer>();
setParents.add(n);
Integer child = n;
while (mapParentByChild.get(child) != null) {
setParents.add(mapParentByChild.get(child));
child = mapParentByChild.get(child);
while (parent != null) {
if (setParents.contains(parent)) {
return true;
parent = mapParentByChild.get(parent);
return false;
2.9k Views
✓ Upvote
            Share Share
                                                                      ♡ ◇ ····
         Add a comment...
                                                     Comment
                                                                 Recommended All
      Vance Faber, Problem solver (2012-present)
                                                                               <u>8</u>
      Answered Jul 3
I have this idea. Use BFS to find a spanning tree keeping track of every edge not
```

on the tree in a list. Each of those edges forms a cycle called a tree cycle and can



Top Stories from Your Feed