

Daffodil International University

Course Code: CSE 214
Course Name: Algorithm Lab

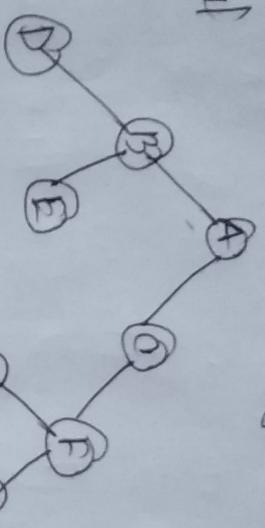
Submitted to
Subroto Mag Pinsky
Department of CSE

Submitted by
Akshandra Mim
Tanjima

ID: 191-15-2455
Sec: 0-14

Topic : 1

Tree is a special kind of cycle graph that contains no cycles for full tree. Mainly we use dfs algorithm for traversal.



At first here we will use time traversal to traversal.

Here, A is the root node.

Here, A is the root node. Then after that, we push A in the stack. After that, we push B and C in the stack.

• So, now the traversal order is : AB

Then, we will check the connected node of B and go one of them. Those two are connected. Then we will go to D and push it into the stack.

► So, the traversal order is now : ABD
Now, there in the top of the stack is D and DFS will check where we can go by D from this tree we can see that D is only connected with B. But we traversed D already. That's why we pop stack. and go back to node B. Then DFS will check unvisited nodes. The only unvisited nodes is E. Then we will go to the node E. Then push E in the stack.

► So, Now the traversal order is ABDB.
Then, we will check the adjacent node of E. E have only one adjacent node and that is B. And then B is already visited.
So, E does not have any adjacent nodes in left side. POP stack and go

back to E. There are no unvisited adjacent nodes of B. So, pop stack again. We came back into nodes A. Now, there is only one adjacent node of A and it is C. That is unvisited. So DPS will go into C. Push C in stack.

- Now, the traversal is ABDEC.

Now, we will check adjacent node of C. There is only one adjacent node and that is F. Now, we push F in the stack.

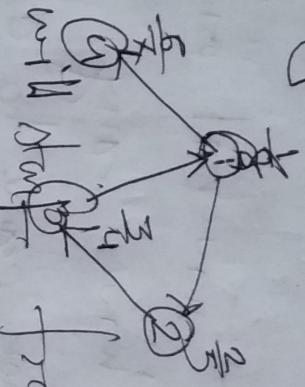
- Now the traversal order is ABCDFC.

Then, we will check adjacent nodes of F. F has two unvisited node that is G and H. Then F chose G first. Then push G into the stack. Then we will go G after that G has no unvisited node so we pop stack.

The traversal order is now ABDFCEFGH
After that we come back into P. F has
met first unvisited node. and it is H push
it into stack. H has no unvisited node
so pop the stack and come back to P.

So, Now the traversal order is ABDFCEFGH
Here, F doesn't have any unvisited node
so, stack pop. Then come to C. C has no
unvisited node. So, again stack pop, then we
come back to A. A haven't any unvisited
nodes. So, again stack pop and the traversal
has stopped

Cycle finding — If there is a back edge in a graph then if there is a back edge in a graph then we can say that tree graph has a cycle.



Let's see what happens from 1 visited. There is no node in 1st node ①. ① is a starting time. Then we will go to 2 as node and the starting time is standing time is 2. Then we will go to 3 as node and here the starting time is 3. No node and here the starting time is 3. Then we can go 3 to 1 as node but 1 as node is already visited. So, we will back to 3. So, now 3 as node ending time is 4. Then we will back into 2 as node. Then the ending time is 5. Then we will go to the

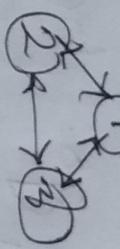
node first is connected with another node and its 4 turn we will go to 4 no nodes. Then we start our starting time is 6. Then, we see that we can't go anywhere from one node. So, here the ending time is 8. Then we go back into node 1 and its ending time is 8.

thus node 3 and node 1 connected with the back edge. So the graph contains a cycle.

Topic No. 3

Component-finding

What we will find out how many sub graph there are in a graph



Now at first in this graph we will check

and run DFS. After 1.DFS if there are any unvisited nodes - then we will run DFS again from that node again. And count.

From this true graph we can find two components.

Topic No: 4

Articulation point finding

Here the main point is that How a separate graph can be found excluding any node or any edge

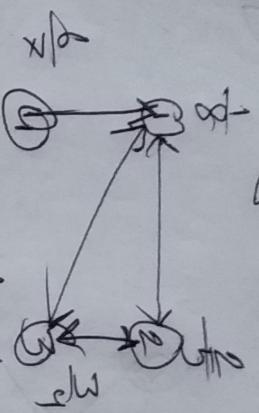


How is this graph we see that 5 node is here. And 3 no node is connected with ① and ② no node and also 3 no node is connected with ④ and ⑤ no node so how if we excluding 3 then we find two graphs. So If we see on articulation point tree two graph that we find from the main graph are ①—② and ④—⑤ but in two two graph has no articulation point. Between here we can't find any node that we can exclude

Topic

Topological Sort

The main theme is how to sort nodes
by descending order of end time.



By applying Johnson's rule we can find the few
swapping times for ending time.

Node:

Starting time	1	2	4	5
ending time	8	5	7	6

Now, Descending order of ending time

8 7 5 6

Now, the node is like that by
following the descending order of ending

point is

1 4 2 3

Here five topological point is

1 4 2 3