## Deepanshu 20196550427

Page:
Date:

Ans=>	Proof by induction
	The a corrected agraph however a veritical rolling with
	any vertex would not leave any vertex as gray (unvesited) and
- 10	all ancestore call return.
	and waters were need there and writers part
-	boute case n=2
	Let two vertices are y and v. without loss of generality, and
	to talled on vi . tollowing events will occur (in order)
	- de comme gellow.
	on ve and ve will be coloured yellow.
3.	Hence, P(2) is true
	Hence, P(2) is true
	ASSESSION 1
	Induction Hypothesis
200	Assume point is true for n=k
(3)	for n- Ktl
1-6	choose any vertex ashitement (and the
200	choose any vertex architectify (call it v). Now DES will be
	will be called next.
	Note that calling DFS on vz cannot ever call DFS on v, since
	colour of v, & changed to yellow before calling the on v.
	Thus, a total of (n+1)-1=n vertices are in consideration.
	Problem effectively reduce to calling Its on n vertical which
	would not leave any vextex as grow (unisited) and all ancestors
	call return according to induction sypothesis.
	Eventually, v, will call return.
	Hence PCK+1) is true whenever PCK) is true.
	Therefore by principle of Modbematical Induction P(n) is true
	for all n.
	Here & Roand

AND

Suppose source is in one component (component) and DFS (x) is called open calling DFS (s) and vis not reachable from s and is in different component (component).

Let DFS is called in following order

s, u, u, u — un v, v2 — vm where

Visve — un ave in component 2.

un. This is possible in 2 cases:

Case-1: Vi is neighbour of un and was coloured gray when DFS(4)

This is not possible as un and v, are in different component:

Case-2: Any one of sur, un - un is ancestore of v.

This is again not possible as v, is in different component
from sur un.

There is no third case possible as DFS is trigged after entering in for loop which require neighbour relationer ancestor relationship.

Here we arrive at a contradiction that & DES on v, connot be called from b, bu, us — un.

Hence, our assumption is wrong and "no recursive talls will be made on nodes not reachable from s".

Hence Proved

Page !

Also note that number of Ages scanned is O(1E1) because—  1. Since every vertex is the visited at most once, the number of edges scanned is roughly truice the number of vertex.  So, in total edges scanned is O(1E1).  The All other operations like checking colour, into changing colour takes constant time.  Hence in total running time is O(v+E) for Dfs.		
the All colouring part taken constant time.  We need to book for pergebour and make recursive was reed to book for pergebour and make recursive calls which take form.  We can say that running fine touthtutury is proportional to number of verticus usuned and number of edges scanned.  Now number of verticus scanned & IVI because—  We visit only state a gray mode, we change its colour to green.  The every time we visit a gray mode, we change its colour to green.  The never change mode of an clour of any mode back to gray also note that number of days scanned is O(IEI) because—  In since every vertex is the visited at most once, the number of edge tour scanned is roughly twice the number of verticus.  The All other operations like checking colour, whose changing colour takes constant time.  Hence in total running time is O(V+E) for Dfs.	Ans Claim: The running time of DES is O(IN+IEI).	1
We can say that running fine contrituting is proportional to number of verticus usuaned and number of edges scanned.  Now number of verticus scanned & IVI because—  We visit only white a gray node.  Every time we visit a gray node, we change its colour to green.  3. We never change node of an colour of any node back to gray Also note that number of does scanned is O(IEI) because—  1. Since every vertex is see visited at most once, the number of edges scanned is roughly twice the number of verticuse, in total edges scanned is O(IEI).  The All other operations like checking colour, where changing colour takes constant time.  Hence in total running time is O(V+E) for D+S.	. It All colouring part takes constant time.	
we can say that running fine wouthtute of is proportional to number of vertices scanned and number of edges scanned.  Now number of vertices scanned & IVI because—  I we visit only write a gray nodes.  2. Every time we visit a gray node, we change its oblows to green.  3. we never change made of an colour of any made back to gray Also note that number of ages scanned is O(IEI) because—  1. Since every vertex is the visited at most once, the number of edges scanned is roughly twice the number of vertices.  So, For total edges scanned is O(IEI).  Then All other operations like heaving colour, whose transfing colours takes constant time.  Hence in total running time is O(V+E) for Dfs.	. We need to look for neighbour and make recursiv	e
we can say that running fine wouthtute of is proportional to number of vertices scanned and number of edges scanned.  Now number of vertices scanned & IVI because—  I we visit only write a gray nodes.  2. Every time we visit a gray node, we change its oblows to green.  3. we never change made of an colour of any made back to gray Also note that number of ages scanned is O(IEI) because—  1. Since every vertex is the visited at most once, the number of edges scanned is roughly twice the number of vertices.  So, For total edges scanned is O(IEI).  Then All other operations like heaving colour, whose transfing colours takes constant time.  Hence in total running time is O(V+E) for Dfs.	calls which take fime.	
Now number of vertices scanned & IVI because —  1. We visit only white a gray node, we change its albour to gray 2. Every time we visit a gray node, we change its albour to gray  3. We never change made of an colour of any made back to gray  Also note that number of Ages scanned is O(IEI) because—  1. Since every vertex is to visited at most once, the number of edge star scanned is roughly twice the number of vertices.  2. The All other operations like checking colour, whose hanging colour takes constant time.  Hence in total running time is O(V+E) for Dfs.		
2. Every time we visit a gray node, we change its colour to green.  3. We never change mode of an colour of any mode back to gray.  Also note that number of alges scanned is O(IE) because—  1. Since every vertex is set visited at most once, the number of edges come scanned is noughly twice the number of vertrus.  20, in total edges scanned is noughly twice the number of vertrus.  1. The All other operations like checking colour, where changing colour takes constant time.  Hence, in total running time is O(V+E) for DfS.	to number of vertices wound and number of edges soun	ed.
2. Every time we visit a gray node, we change its colour to green.  3. We never change made of an colour of any mode back to gray Also note that number of alges scanned is O(1E) because—  1. Since every vertex is sets visited at most once, the number of edges come scanned is noughly twice the number of vertrus.  So, its total edges scanned is noughly twice the number of vertrus.  Then All other operations like checking colour, when changing colour takes constant time.  Hence, in total running time is O(vtE) for DfS.	Now number of vertices scanned < IVI because -	
Also note that number of alger scanned is O(IEI) because—  1. Since every vertex is the visited at most once, the number of edges scanned is O(IEI) because—  1. Since every vertex is the visited at most once, the number of vertron-  2. In total edges scanned is O(IEI).  Then All other operations like checking colour, orders changing colour takes constant time.  Hence, in total running time is O(V+E) for Dfs.	1. We visit only white a gray nodes.	
Also note that number of edges scanned is O(IE) because—  Since every vertex is the visited at most once, the number of edges was scanned is roughly twice the number of vertrus.  So, no total edges scanned is acceptly twice the number of vertrus.  The All other operations like checking colour, whose changing colour takes constant time.  Hence, in total running time is O(V+E) for DfS.	2. Every time we visit a gray node, we change its colow	z to
since every vertex is the visited at most once, the number of edges scanned is roughly twice the number of vertex.  So, no total edges scanned is QIED.  Then All other operations like checking colour, enters changing colour takes constant time.  Hence, in total running time is O(v+E) for Dfs.	3. We never change node of an colour of any node back to	gray
So, in total edges scanned is O(1E1).  Here All other operations like checking colour, whose tranging colour takes constant time.  Hence in total running time is O(v+E) for Dfs.	Also note that number of ages scanned is O(IE) because	2-
So, in total edges scanned is O(1E1).  Here All other operations like checking colour, whose tranging colour takes constant time.  Hence in total running time is O(v+E) for Dfs.	1. Since every vertex is see visited at most once the	1400
Hence on total running time I O(v+E) for Dfs.	of edge scanned is roughly twice the number of ver	tPcu.
Hence en total running time is O(v+E) for Dfs.		
Hence in total running time is O(v+E) for Dfs.	colour takes constant time.	79
There is the work of the state	74410 4000 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	
Henry many description be consorted and "no recented to " or recented to "	There in told sunning time 4 O(v+E) for Dfs.	
teaths will be made as redu at readquale fearurs."	the state of the s	
cally will be readle on willy for reading	of the could be a property of the could be and the could be and the could be a second to the could be a second be	
	there is a straight of readingle from so	No.
January V	a liver alling	
	Have Pereg	