Problem

Show that if P = NP, a polynomial time algorithm exists that takes an undirected graph as input and finds a largest clique contained in that graph. (See the note in Problem 7.38.)

Step-by-step solution

Step 1 of 2

Class - *P* : P is a class of Languages that are decidable in polynomial time on a deterministic single tape Turing machine.

Class - *NP:* NP is a class of Languages that are decidable in polynomial time on a nondeterministic Turing machine.

Clique: A clique in an undirected graph is a sub graph, wherein every two nodes are connected by an edge.

• If $P = NP$ then we have to show that "a polynomial time algorithm exists that takes an undirected graph as input and finds the largest clique in the graph.

• A k – clique is a clique that have k-nodes.

• We know that "clique is in NP".

• So if $P = NP$ then clique is in P.

Therefore, if $P = NP$ then clique is recognizable in polynomial time.

Comment

Step 2 of 2

The following algorithm will find the largest clique in the graph:

1. Let n be the no. of nodes in the given graph G.

i be the variable which runs from 1 ton.

2. Using the polynomial time algorithm form clique, check whether there exist a clique of size i.

3. Output the Large i for which a clique exists.

To find the maximum clique, we start with i, the maximum clique size.

Remove one node and see if there is still a clique of size i.

If not, restore that node and remove another node.

If so, respect the process until we are left with a graph of i nodes, which must be a clique.

• This algorithm will take almost n trials to find which node to remove and at most n nodes to be removed.

Then the total running time is polynomial.

Comment