NAME : SUBHAPREET PATRO
ROLL NO. : 2211CS010547
CLASS : GROUP-7A

14/10/23

DAA ASSIGNMENT - 2

1) Write the Union and Find Algorithms with suitable example and analyze its complexity.

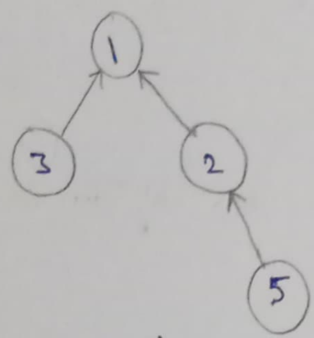
Ans. • <u>Simple Algorithm for Find (i) :</u>

```
Algorithm Find (i) // Determines the root of the tree containing element i.
{
while (p[i] >= 0)
do i := p[i];
return i;
}
```

• <u>Collapsing rule for Find (i) :</u>

```
Algorithm CollapsingFind (i)
// Find the root of the tree containing element i. Use the collapsing rule
// to collapse all nodes from i to the root.
{
    r := i;
    while (p[r] > 0) do r := p[r]; // Find the root
    while (i ≠ r) do // Collapse nodes from i to root r
    {
        s := p[i]; p[i] := r; i := s;
    }
    return r;
}
```

Ex:



Array Representation

i	1	2	3	5
$P[i]$	-1	1	1	2

Find (5) = 1

Find (2) = 1

Find (3) = 1

The root node represents all the nodes in the tree.

- Simple Algorithm for Union

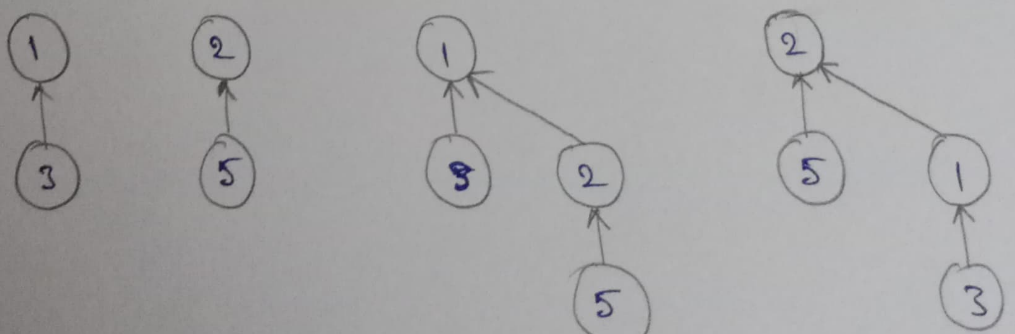
```
Algorithm Union (i, j)
{
P[i]:= j;
}
```

- **Weighting rule for Union (i,j) :-**

Algorithm WeightedUnion (i,j)
// Union sets with roots i and j, i ≠ j, using the weighting rule.
// p[i] = - count [i] and p[j] = - count [j].

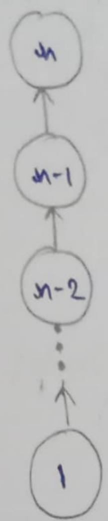
```
{
    Temp := p[i] + p[j];
    if (p[i] > p[j]) then
    { // i has fewer nodes.
        p[i] := j; p[j] := Temp;
    }
    else
    {// j has fewer or equal nodes.
        p[j] := i; p[i] := Temp;
    }
}
```

Ex:- Implement following sequence of operations :- Union (3,1), Union (5,2), Union (2,1) and Union (1,2).



• **Analysis:**

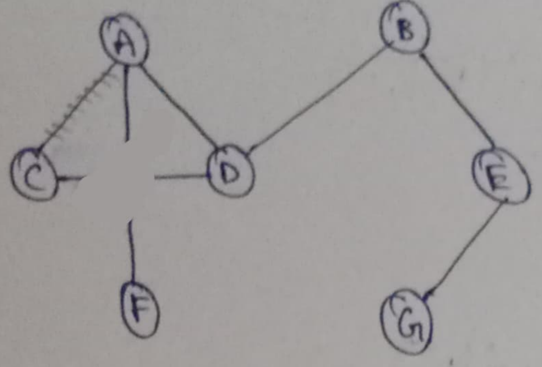
The sequence of Union operations results the degenerate tree as below.



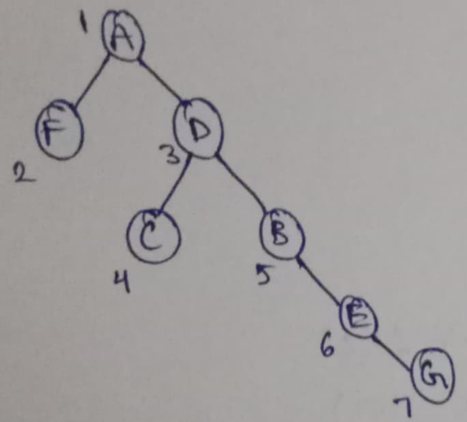
Since, the time taken for a Union is constant, the $n-1$ sequence of union can be processed in time $O(n)$; and for the sequence of Find operations it will take time complexity of $O\left(\sum_{i=1}^{n} i\right) = O(n^2)$.

2) Calculate the articulation points for the following graph and draw the bi-connected components.

Ans.



Depth First Spanning Tree :-



• Calculation of $L(u)$:-

$$\boxed{L(u) = \min\{dfn(u), \min[L(w)], dfn[L(w)]\}}$$

$L(G) = \min\{7, -, -\} = 7$

$L(C) = \min\{4, -, -\} = 4$

$L(F) = \min\{2, -, -\} = 2$

$L(E) = \min\{6, \min[L(G)], -\}$

$= \min\{6, 7, -\}$

$= 6$

$L(B) = \min\{5, \min[L(E)], -\}$

$= \min\{5, 6, -\}$

$= 5$

$L(D) = \min\{3, \min[L(C), L(B)], -\}$

$= \min\{3, \min[4, 5], -\}$

$= \min\{3, 4, -\}$

$= 3$

$L(A) = \min\{1, \min[L(F), L(D)], -\}$

$= \min\{1, \min[2, 3], -\}$

$= \min\{1, 2, -\}$

$= 1$

- Checking articulation points :-

$\boxed{dfn(u) \leq L(w)}$

Check node 'D' :- $dfn(D) \leq L(B), L(C)$

$3 \leq 4, 5$ (✓)

Check node 'B' :- $dfn(B) \leq L(E)$

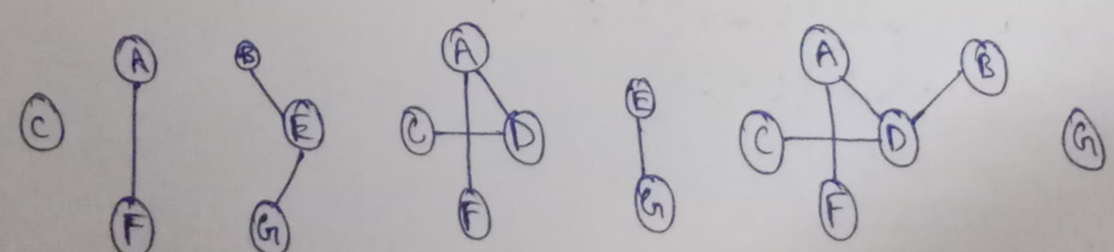
$5 \leq 6$ (✓)

Check node 'E' :- $dfn(E) \leq L(G)$

$6 \leq 7$ (✓)

$\therefore$ Articulation points :- D, B, E

- Biconnected components :-



3) Define the following terms: state space, explicit constraints, implicit constraints, problem state, solution states, answer states, live node, E-node, dead node, bounding functions.

Ans.

- State space :- The state space of a dynamical system is the set of all possible states of the system. Each coordinate is a state variable, and the values of all the state variables completely describes the state of the system. Representation of a state space as a tree is called as a state space tree.

- Explicit constraints :- It is a rule in which how element in a tuple is related.
- Implicit constraints :- The rules that restrict each element to be chosen from the given set.
- Problem state :- It is each node in the depth first search tree
- Solution states :- They are the problem states 'S' for which the path from the root node to 'S' defines a tuple in the solution space.
- Live node :- It is a node that has been generated but whose children have not yet been generated
- E-node :- It is a live node whose children are currently being explored. In other words, an E-node is a node currently being expanded.

- **Dead node:** It is a generated node that is not to be expanded or explored any further. All children of a dead node have already been expanded.

- **Bounding functions:** Let B be a function from the set of vertices of the state space tree to the positive integers. Suppose that for any partial solution x, $B(x) \geq P(x)$ then we say B is a ~~founding~~ bounding function.

4) Write the Backtracking algorithm for solving N-Queens problem with state space tree.

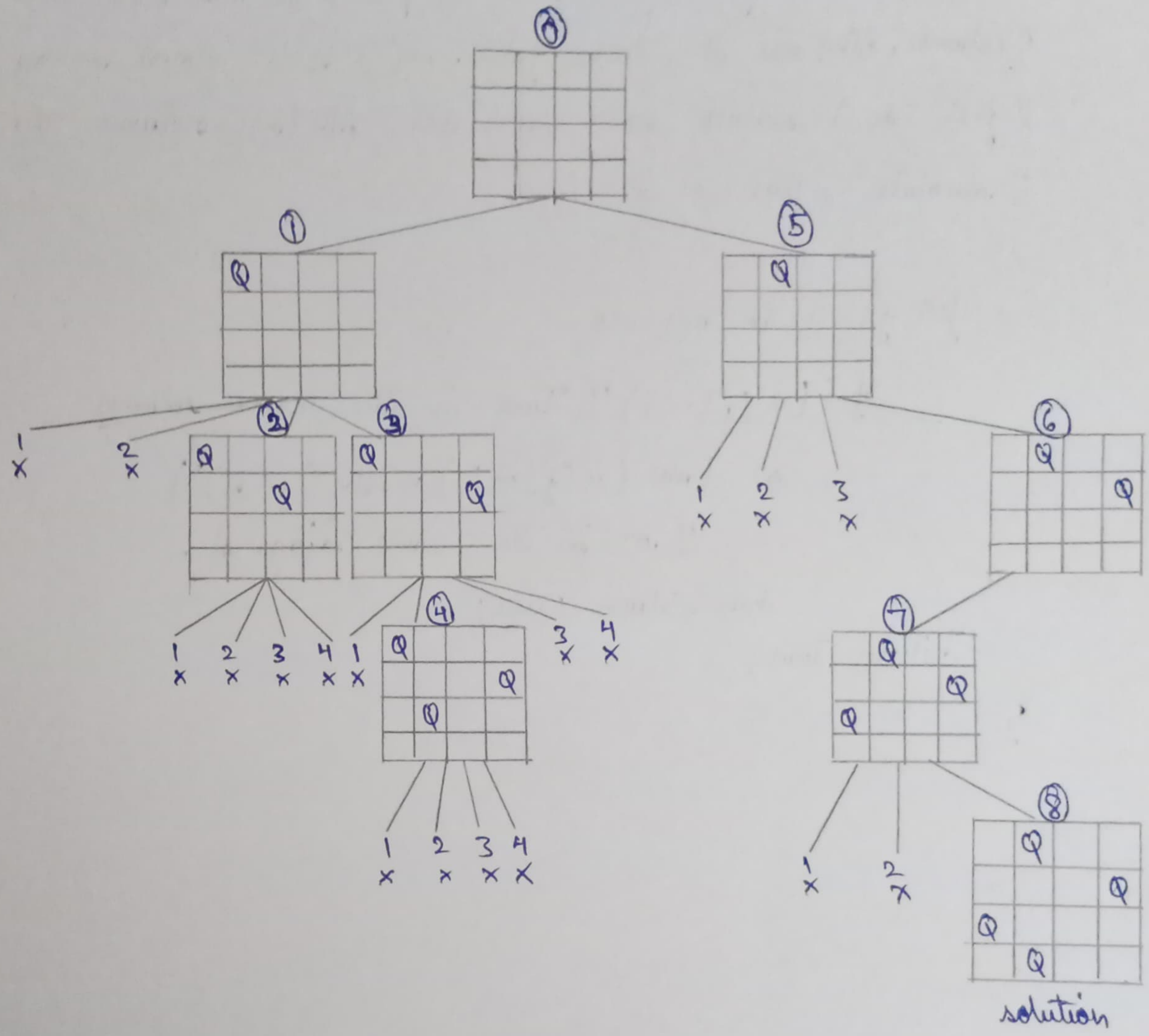
Ans. Algorithm NQueens (k, n)

```
// Using backtracking, this procedure prints all possible placements of
// n queens on an n x n chessboard so that they are not
// attacking each other.
{
    for i := 1 to n do
    {
        if Place (k, i) then
        {
            x[k] := i;
            if (k = n) then write (x[1:n]);
            else NQueens (k+1, n);
        }
    }
}
```

Algorithm Place (k,i)
// Returns true if a queen can be placed in kth row and ith
// column. Otherwise it returns false. x[ ] is a global array whose
// first (k-1) values have been set. Abs (r) returns the
// absolute value of r.

```
{
    for j := 1 to k-1 do
        if ((x[j] = i) // Two in the same column
            or (Abs (x[j] - i) = Abs (j - k)))
                // or in the same diagonal.
            then return false;
    return True;
}
```

State space tree for 4- Queens problem:

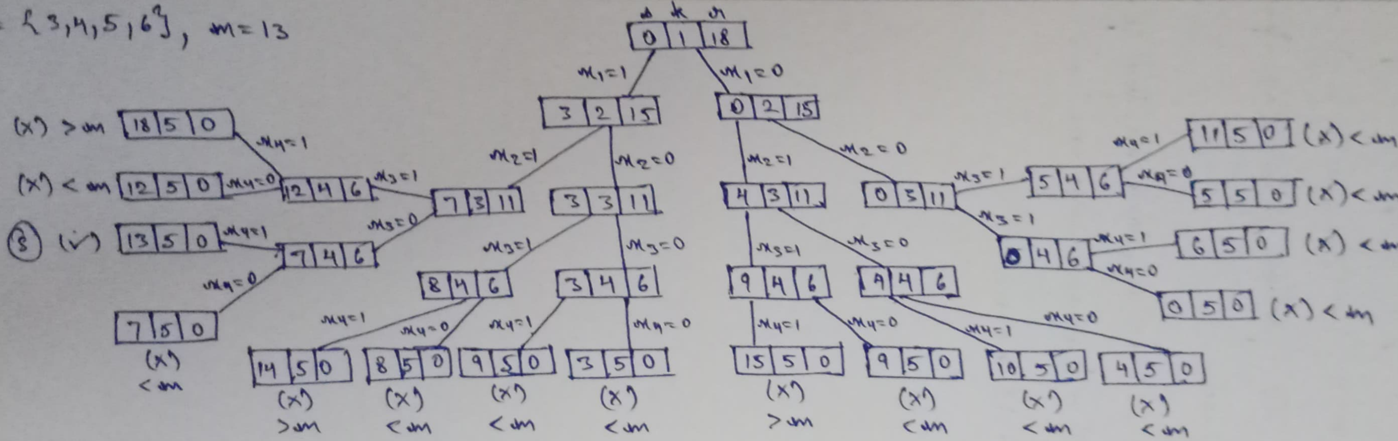


5) Give the statement of ~~sum subset of~~ sum of subset problem. Find all sum of subsets for $n=4$, $(w1, w2, w3, w4) = (3,4,5,6)$ and $M=13$. Draw the portion of the state space tree using fixed - tuple sized approach.

Ans. → Given positive numbers w_i, $1 \le i \le n$, and m, this problem requires finding all subsets of w_i whose sums are 'm'.

→ All solutions are k-tuples, $1 \le k \le n$.

$S = \{3, 4, 5, 6\}, \; m = 13$



$\therefore$ Solution Set = $\{3, 4, 6\} \Rightarrow w_1 = 1$

$w_2 = 1$

$w_3 = 0$

$w_4 = 1$