CS & IT ENGINEERING

Algorithm

Miscellaneous

Lecture No. [redacted] 04

By- Aditya Jain sir

Recap of Previous Lecture

Topic

Topic

Topic

TC analysis using

Recursion Tree Appr

Topics to be Covered

Topic

Topic

Topic

Set
- union
- Find
- Kruskal

About Aditya Jain sir

1. Appeared for GATE during BTech and secured AIR 60 in GATE in very first attempt - City topper
2. Represented college as the first Google DSC Ambassador.
3. The only student from the batch to secure an internship at Amazon. (9+ CGPA)
4. Had offer from IIT Bombay and IISc Bangalore to join the Masters program
5. Joined IIT Bombay for my 2 year Masters program, specialization in Data Science
6. Published multiple research papers in well known conferences along with the team
7. Received the prestigious excellence in Research award from IIT Bombay for my Masters thesis
8. Completed my Masters with an overall GPA of 9.36/10
9. Joined Dream11 as a Data Scientist
10. Have mentored 12,000+ students & working professions in field of Data Science and Analytics
11. Have been mentoring & teaching GATE aspirants to secure a great rank in limited time
12. Have got around 27.5K followers on Linkedin where I share my insights and guide students and professionals.

@ADITYASIR_PW

Telegram Link for Aditya Jain sir: https://t.me/AdityaSir_PW

Topic: Sets

Unordered

~~Wonder~~ collection of non- repeated/ non- duplicate/ distinct elements.

Example:

$S_1 = \{1, 5, 7, 2\}$

There ~~is~~ number including in sets.

5

7 1

2

Topic: Sets

Veen diagram:

1. Disjoint sets: Mutually exclusive sets
   ~~number~~ no common elements

~~Equation:~~ Eg :-

S1: set of even numbers = {2, 4, 6}

S2: set of ~~even~~ odd numbers = {1, 3, 5}

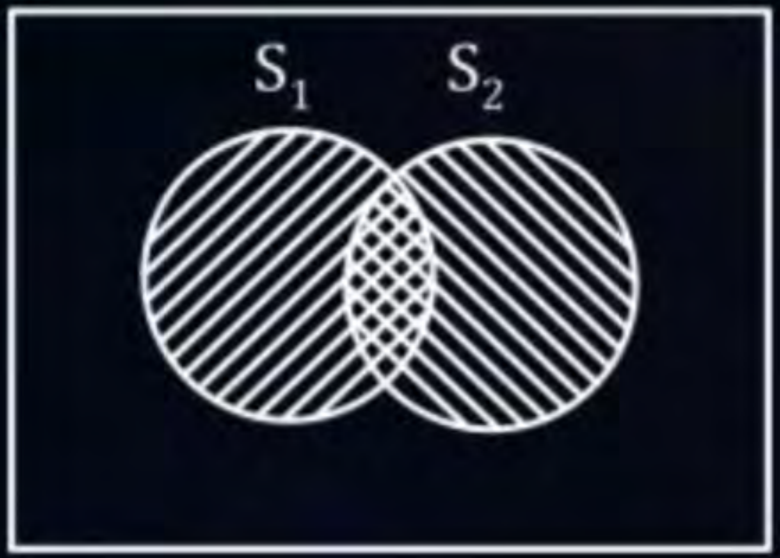
S1 $\cap$ S2 = ϕ (empty set)

$|S1 \cap S2| = 0$

Topic: Sets

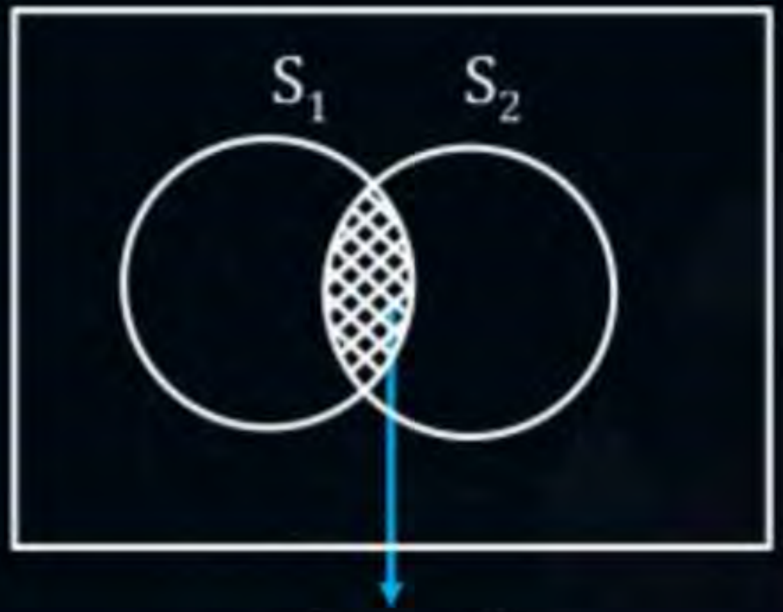
1. Union:

$S_1 \cup S_2 = \{1, 2, 3, 4, 5, 6\}$



2. Intersection

$S_1 \cap S_2$



$S_1 \cap S_2$

Prev eg:

$S1 \cap S2 = \phi$

3. Disjoint sets

$S_1 \cup S_2 = \{1, 2, 3, 4, 5, 6\}$

Topic: Sets

3. Set Difference: $S_1 - S_2$

Element in S1 that are not is S2.

Equation:

S_1: {2, 3, 4}

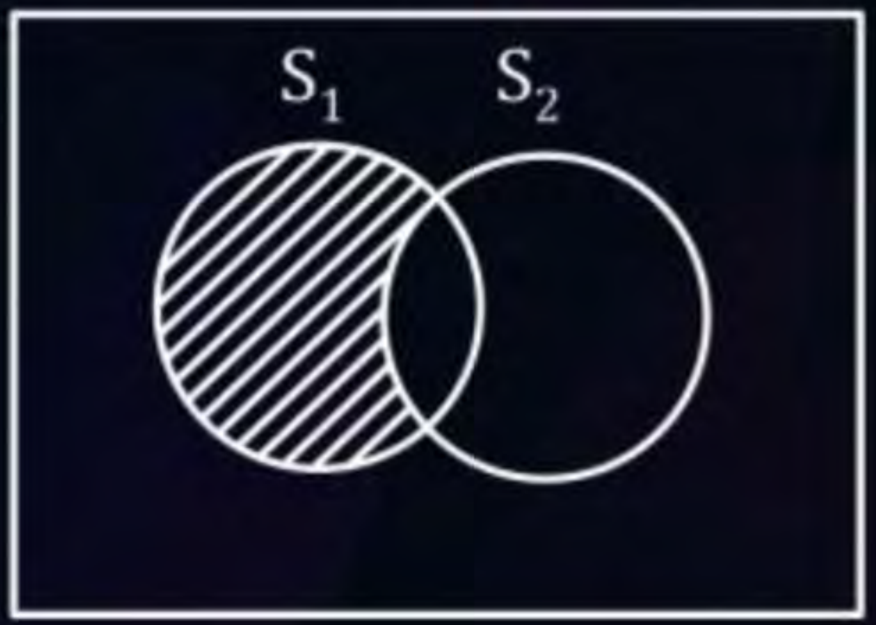
S_2: {3, 5, 7}

$$S_1 - S_2 = \{2,3,4\} - \{3\}$$
$$= \{2, 4\}$$

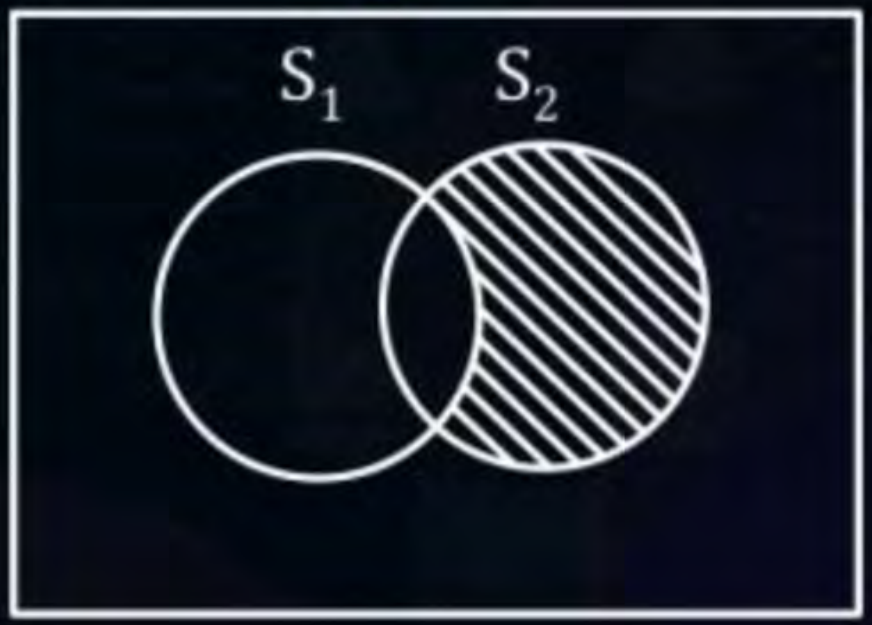
Is $S_1 - S_2 = S_2 - S_1$

Not necessary

$S_1 - S_2$



$S_2 - S_1$

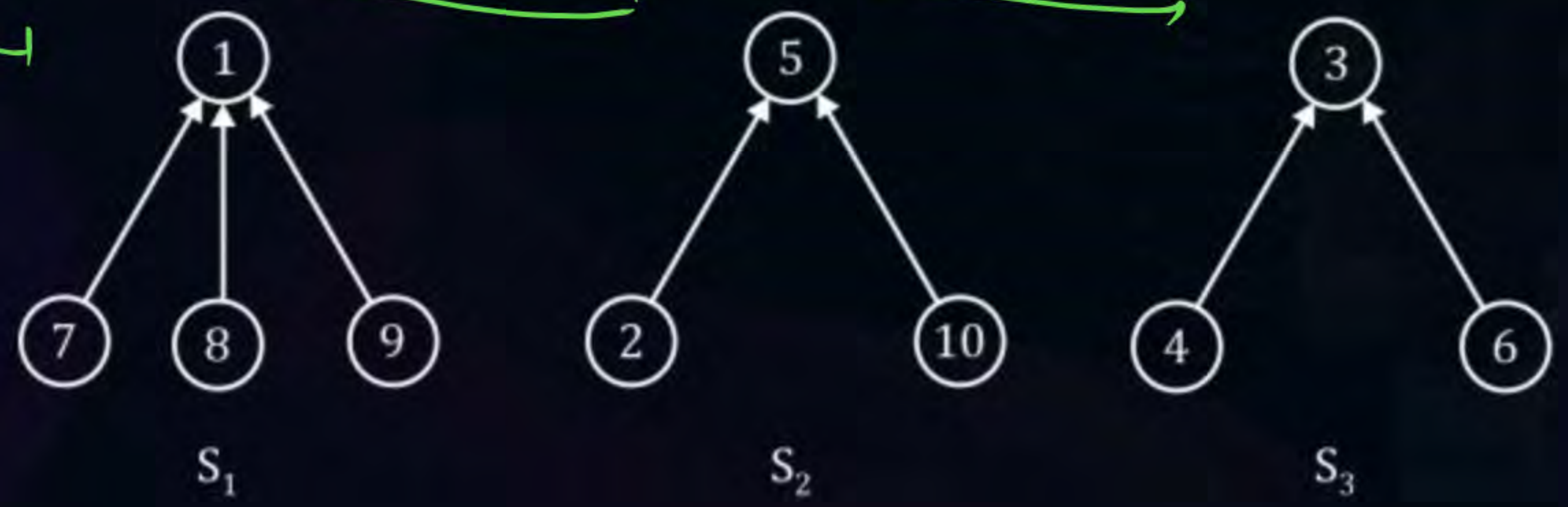


Topic: Sets

Algo

It is assumed that the elements of the sets are the numbers 1, 2, 3, n. These number might, in practice, be indices into a symbol table in which the names of the elements are stored. We assume that the sets being represented are pairwise disjoint (that is, if S_i, and S_j, $i \neq j$, are two sets, then there is no element that is in both S_i and S_j).

For example, when n = 10 the elements can be partitioned into three disjoint sets, $S_1 = \{1, 7, 8, 9)$, $S_2 = \{2, 5, 10\}$, and $S_2 = \{3, 4, 6\}$. representation for these



Important points:

1. In tree diagram representation of a set, the Root of the tree is considered as the representative of the set.
2. Usually (not mandatory), that the 1st element of the set is the representative of that set.

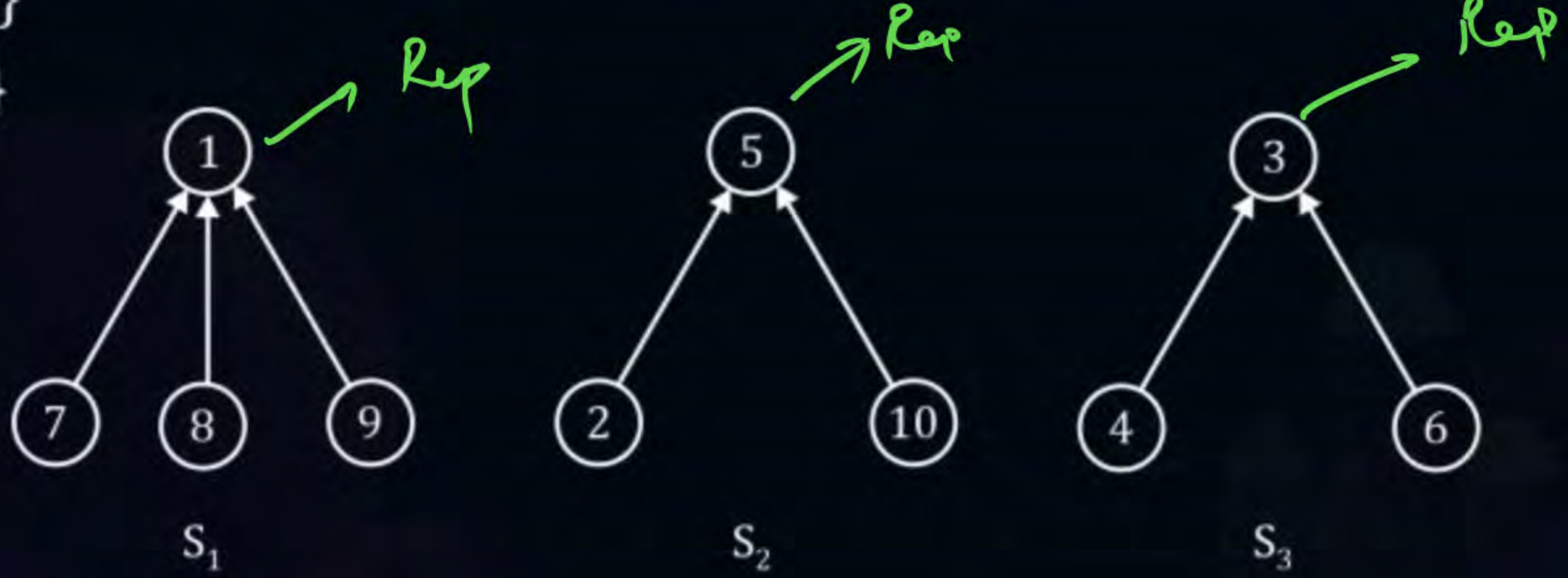
Topic: Sets

Tree Diagram Representation of a set:

S_1 : $\{1, 7, 8, 9\}$

S_2 : $\{2, 5, 10\}$

S_3 : $\{3, 4, 6\}$



Find (8) = 1 Find (7) = 1

Find (1) = 1 Find (10) = 5

Find (3) = 3

Topic: Sets

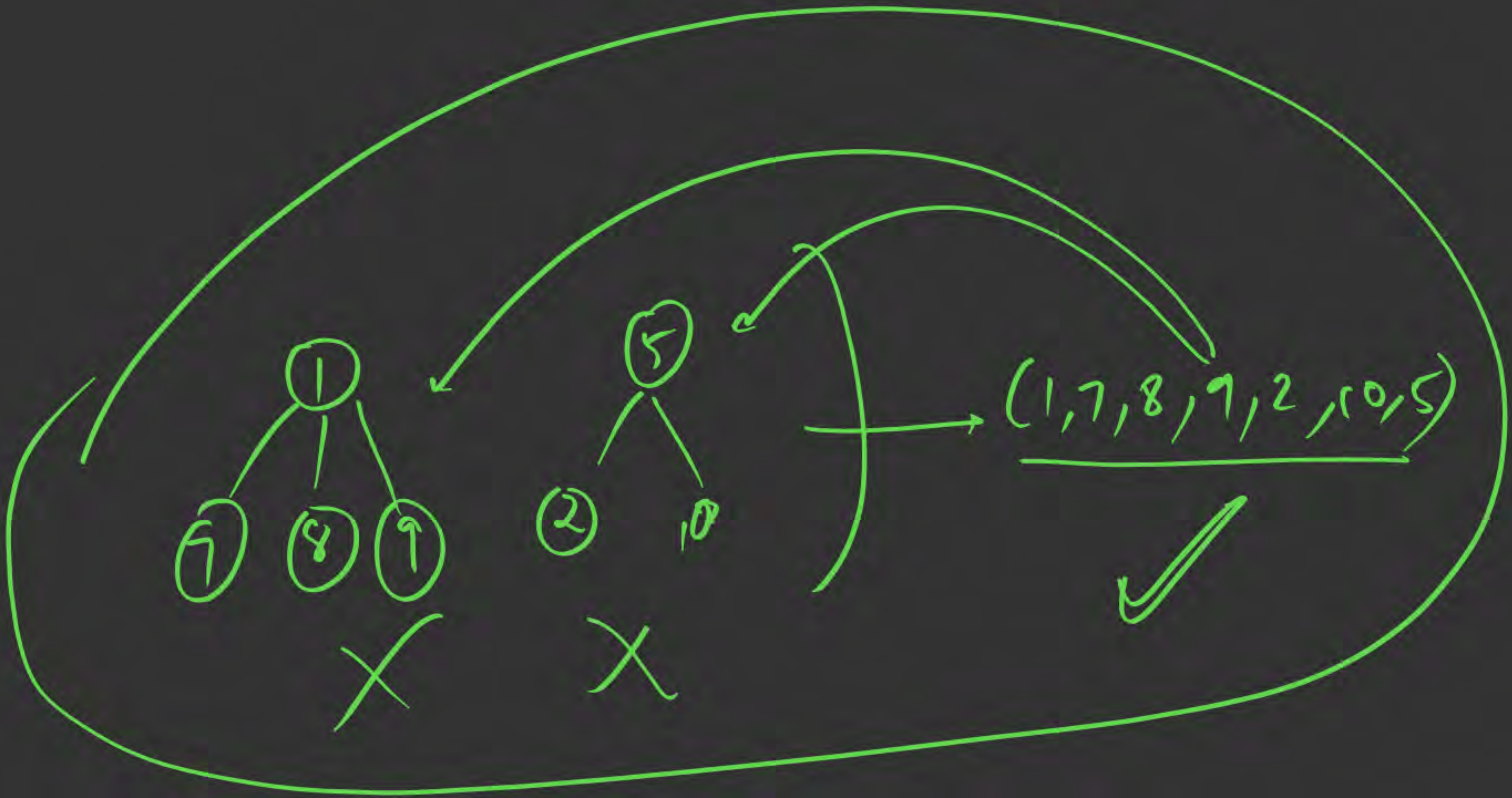
V.V.imp

The operations we wish to perform on these sets are:

1. Disjoint set union. If S_i and S_j are two disjoint sets, then their union $S_i \cup S_j$ = all element x such that x is in S_i or S_j. Thus $S_1 \cup S_2 = \{1, 7, 8, 9, 2, 5, 10)$. Since we have assumed that all sets are disjoint, we can assume that following the union of S_i and S_j the sets S_i and S_j do not exist independently; that is, they are $\boxed{\text{replaced}}$ by $S_i \cup S_j$ in the collection of sets.
2. Find (i). Given the element i, find the set containing i. Thus, 4 is in set S_3, and 9 in set S_1.

↓

Returns Representative of that set.


1
7 8 9
5
2 10
(1,7,8,9,2,10,5)

Topic: Sets

Party-1 → R1

union

Party-2 → R2

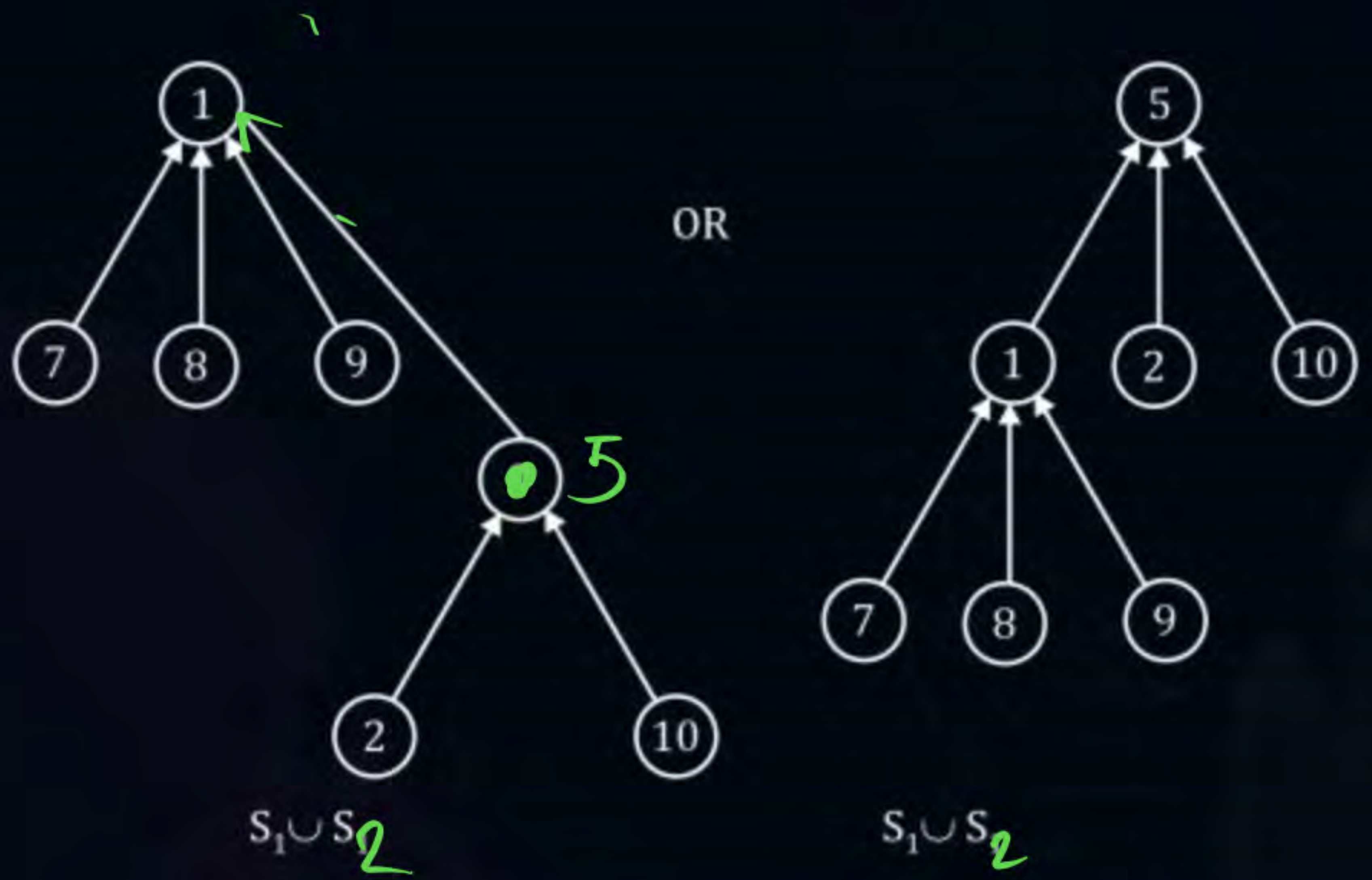
Party-1 + Party-2

R1 or R2

Topic: Sets

$S_1 : \{1, 7, 8, 9\}$

$S_1 : \{5, 2, 10\}$



Possible representation of $S_1 \cup S_2$

Topic: Sets

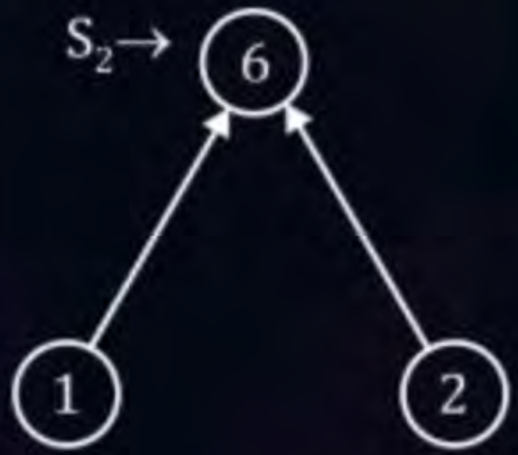
Imp

Important:

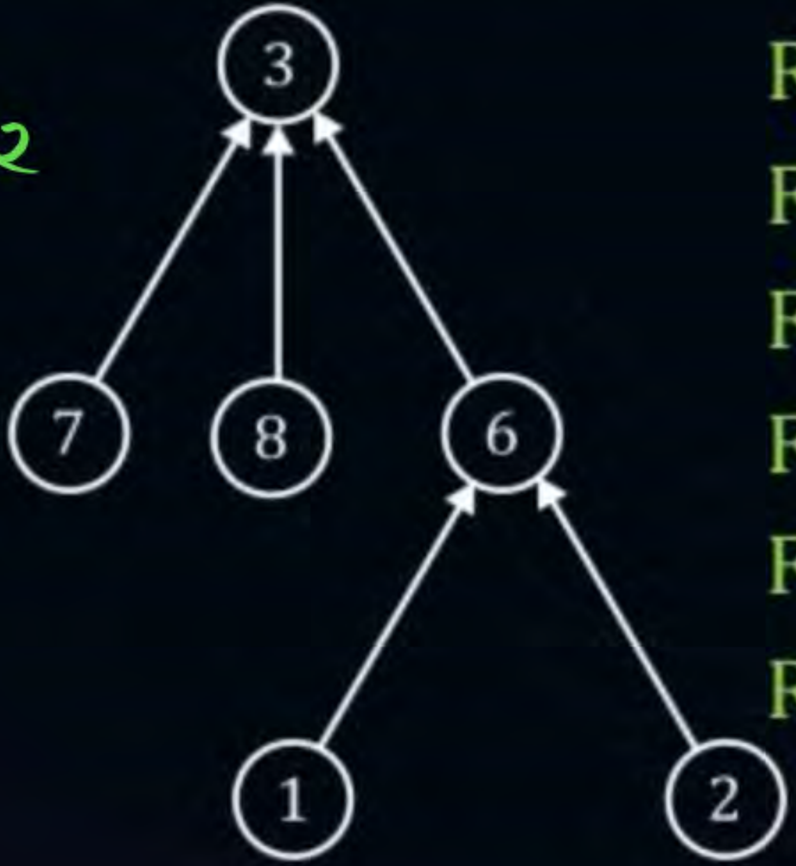
Difference between parent and find/ Representative

S_1: {3, 7, 8}

S_1: {6, 1, 2}



$S_1 \cup S_2$



Representative

Find (8) = 3

Find (6) = 3

Find (2) = 3

Find (i) always gives the root

Representative

Topic: Sets

Important Role:

In above $S_1 \cup S_2$

6 is the parent of 2 but 3 is the overall set representative.

Parent (2) = 6

$\boxed{\text{Find (2) = 3}}$

Parent (Root) = –1

Parent (3) = –1
Find (3) : 3

Topic: Sets

Parent (Representative) = –1

Parent (Root) = –1

i	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
P	–1	5	–1	3	–1	3	1	1	1	5

Array representation of S_1, S_2 and S_3 of figure

S_1: {1, 7, 8, 9}

S_2: {2, 5, 10}

S_3: {3, 4, 6}

[NAT]

#Q. Write down the array representation of the given tree diagram of set.

Topic: Sets

Solution:

i	1	2	3	4	5	6	7	8	9	10
Parent [i]	5	5	5	10	–1	1	1	2	10	2

Find (4) = 5

Parent (4) =10

Find (5) = 5

~~Find~~ Parent (5) = –1

Topic: Sets

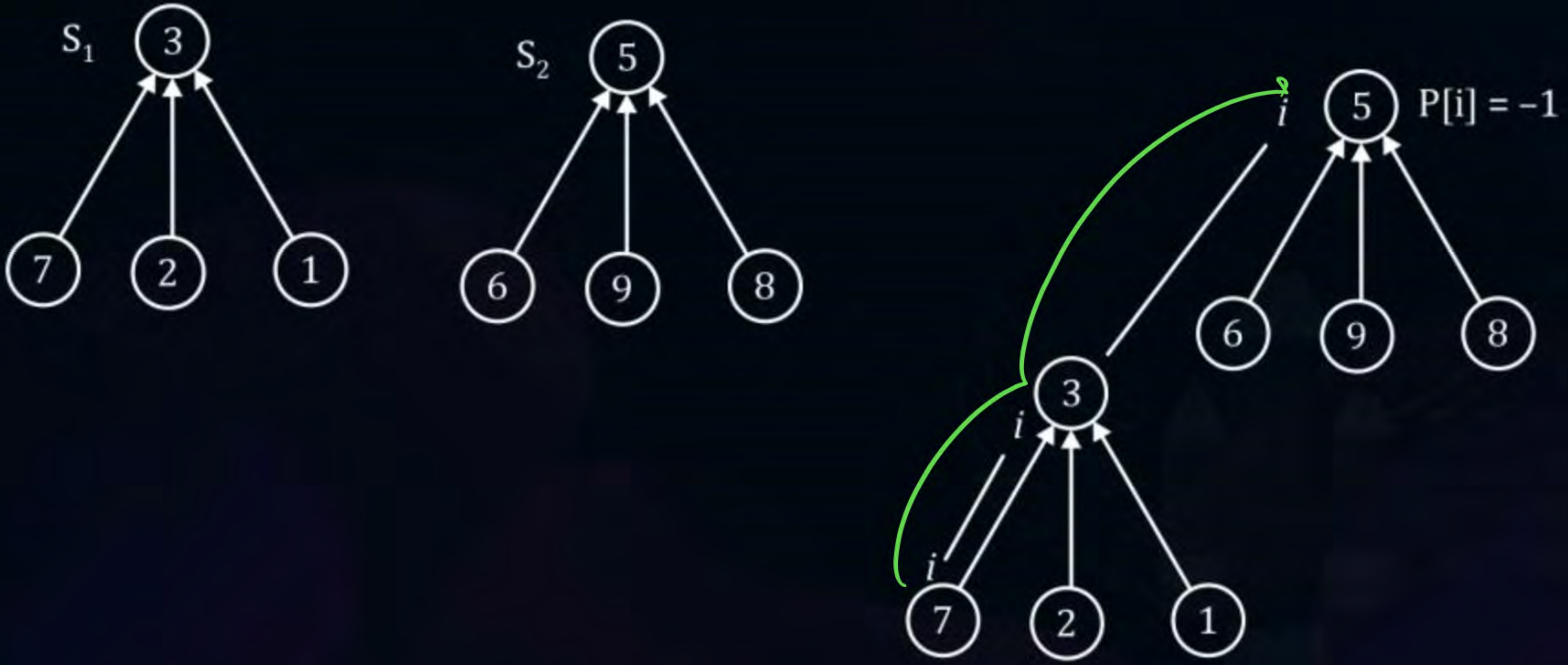
Imp

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

```
Algorithm find (i)
{
    While (p [i]≥ 0) do i=p [i];
    return i
}
```

$p(i) = -1$

Topic: Sets



Topic: Sets

$i = 7$

Find (7)

$i = 7\ P[i] = 3 \geq 0$

$i = 3\ P[3] = 5 \geq 0$

$i = 5\ P[5] = -1 \geq 0$ (False)

exit loop

return (5)

Topic: Greedy Method

temp

```
Algorithm Kruskal [E, cost, n, t]
{
Construct a min Heap out of the edge cost using Heapify
For i = 1 to n do parent [i] : = –1;
i : 0; mincost: = 0.0;
While ((i< n –1) and (heap not empty)) do
{
Delete a minimum cost edge (u,v) from the heap and reheapify using adjust;
J = find (u);k = find (v);
If (j≠k then)
{
i: i + 1
```

cycle avoidance

i = i+1

Topic: Greedy Method

```
T[i,1]: =u; [I, 2 ] =v
Mincost := mincost + cost [u, v]
     Uninon (j ,k)
     }
}
     If (i ≠ n−1) then write ("No spanning tree")
     Else return mincost;
}
```

Topic: Sets

Code walk through:

Given graph: edge Cost: [~~9~~, ~~11~~, ~~13~~, ~~15~~, ~~17~~, ~~21~~, ~~23~~, ~~24~~, 27]

Min heap of edge cots of given graph.

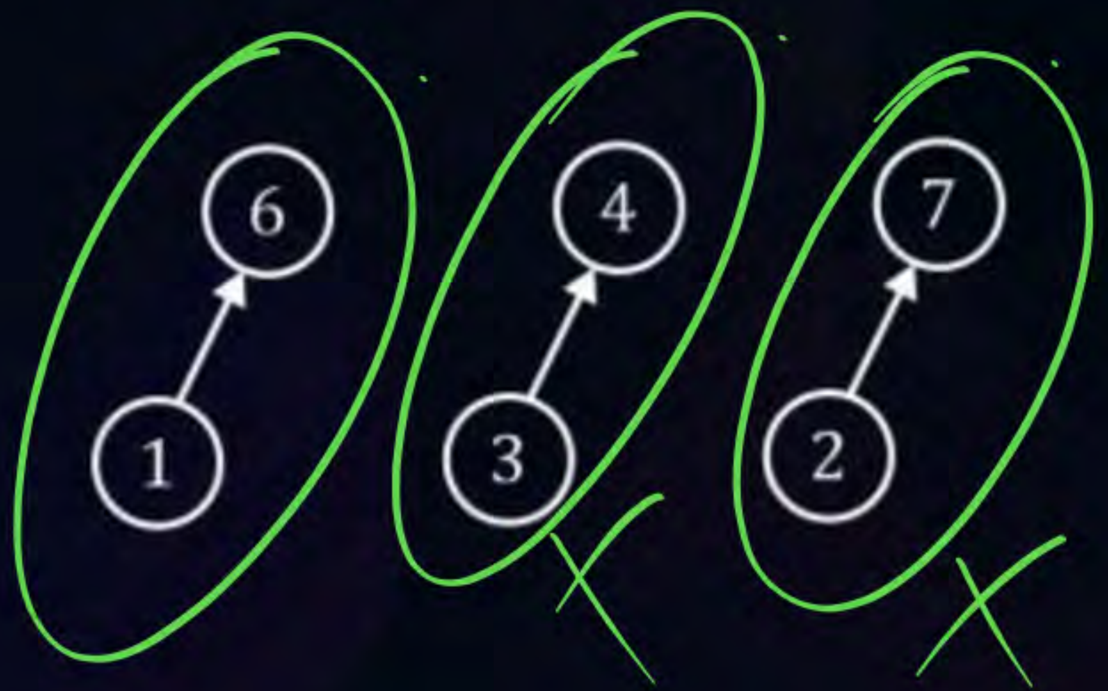
skip

Topic: Sets

Initially

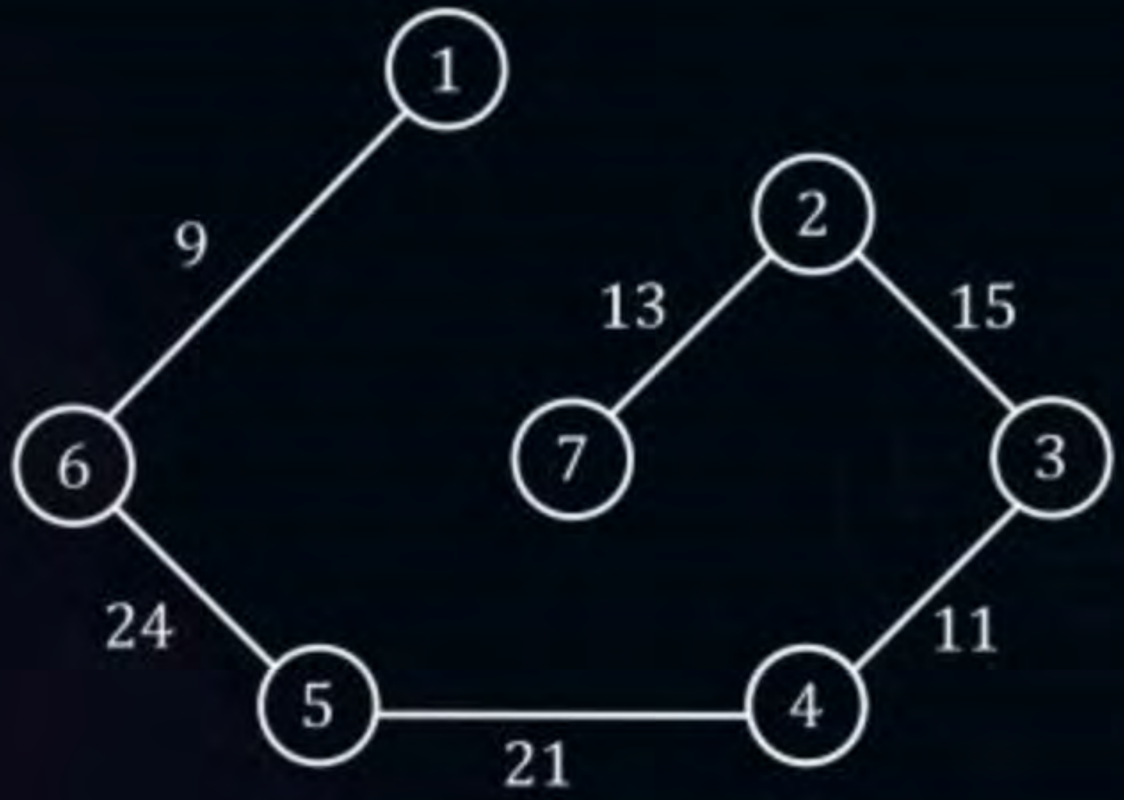
Parent:

-1	-1	-1	-1	-1	-1	-1
1	2	3	4	5	6	7



Topic: Sets

Min cost: $= \underline{0 + 9} + 11 + 13 + 15 + 21 + 24 = 93$



Topic: Sets

1. $(u, v) = (1,\ 6)$

$u \quad v$

$j = find\ (1) = 1$ ✓

$k = find\ (6) = 6$ ✓

$j \neq k$

$t\ [1, 1]\ = 1$

$t\ [1, 2]\ = 6$

Union $(1, 6)$

2. $(u, v) = (3,\ 4)$

$j = find\ (3) = 3$

$k = find\ (4) = 4$

$j \neq k$

$t\ [2, 1]\ = 3$

$t\ [2, 2]\ = 4$

Union $(3, 4)$

Topic: Sets

3. $(u, v) = (2, 7)$

$j = \text{find}(2) = 2$ ✓
$k = \text{find}(7) = 7$ ✓

$j \neq k$
t [3, 1] =2
t 3, 2] =7
Union (2, 7)

4. $(u, v) = (2, 3)$

$j = \text{find}(2) = 7$ ✓
$k = \text{find}(3) = 4$ ✓

$j \neq k$
t [4, 1] = 2
t [4, 2] = 3
Union (7, 4)

Topic: Sets

5. $(u, v) = (7, 4)$

$j = \text{find}(7) = 4$

$k = \text{find}(4) = 4$

$j = k$

(cycle)

6. $(u, v) = (5, 4)$

$j = \text{find}(5) = 5$ ✓

$k = \text{find}(4) = 4$ ✓

$j \neq k$

$t[5, 1] = 5$

$t[5, 2] = 4$

Union $(5, 4)$

Topic: Sets

H.W

7. $(u, v) = (7,\ 5)$

$j = \text{find}(7) = 4$

$k = \text{find}(5) = 4$

$j = k$

(cycle)

8. $(u, v) = (5,\ 6)$

$j = \text{find}(6) = 4$

$k = \text{find}(5) = 4$

$j \neq k$

$t[6, 1] = 6$

$t[6, 2] = 5$

Union $(6, 4)$

Topic: Sets

Time and space complexity analysis of Kruskal (overall).

e = number of edges in given graph

1. Time complexity $= \boxed{O(e^* \log e)}$
2. Space complexity $= O(e + v)$

$= \boxed{O(e + n)}$

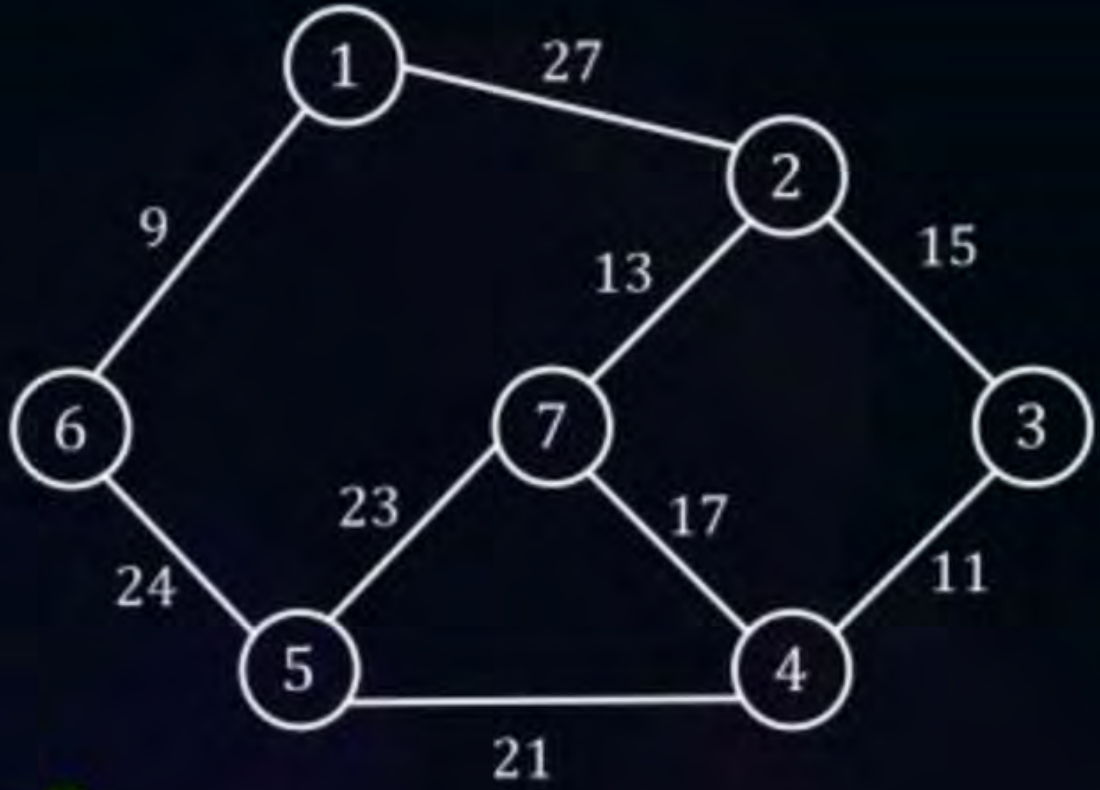
$|V| = n$

$|E| = e$

Topic: Sets

Kruskal logic:

Given a graph:



$n = |v| = 7$

MCST edges $= n - 1$

$= 6$

Cost of MCST

$= 9 + 11 + 13 + 15 + 21 + 24 = 93$

stop

MCST

[MCQ]

#Q. Consider a max heap, represented by the array: 40, 30, 20, 10, 15, 16, 17, 8, 4

Array Index	1	2	3	4	5	6	7	8	9
Value	40	30	20	10	15	16	17	8	4

Now consider that a value 35 is inserted into this heap. After insertion, the new hope is

HW

A. 40, 30, 20, 10, 15, 16, 17, 8, 4, 35

B. 40, 35, 20, 10, 30, 16, 17, 8, 4, 15

C. 40, 30, 20, 10, 35, 16, 17, 8, 4, 15

D. 40, 35, 20, 10, 15, 16, 17, 8, 4, 30

2 mins Summary

Set

THANK - YOU