Recursion Part 2

Tree Recursion

⇒ Input [1,2,3] ⇒ Create All the Subsets of this list

⇒ Subsets ⇒ [[], [1], [2], [3]
[1,2], [1,3], [2,3]
[1,2,3]]

⇒ N ⇒ 2^N

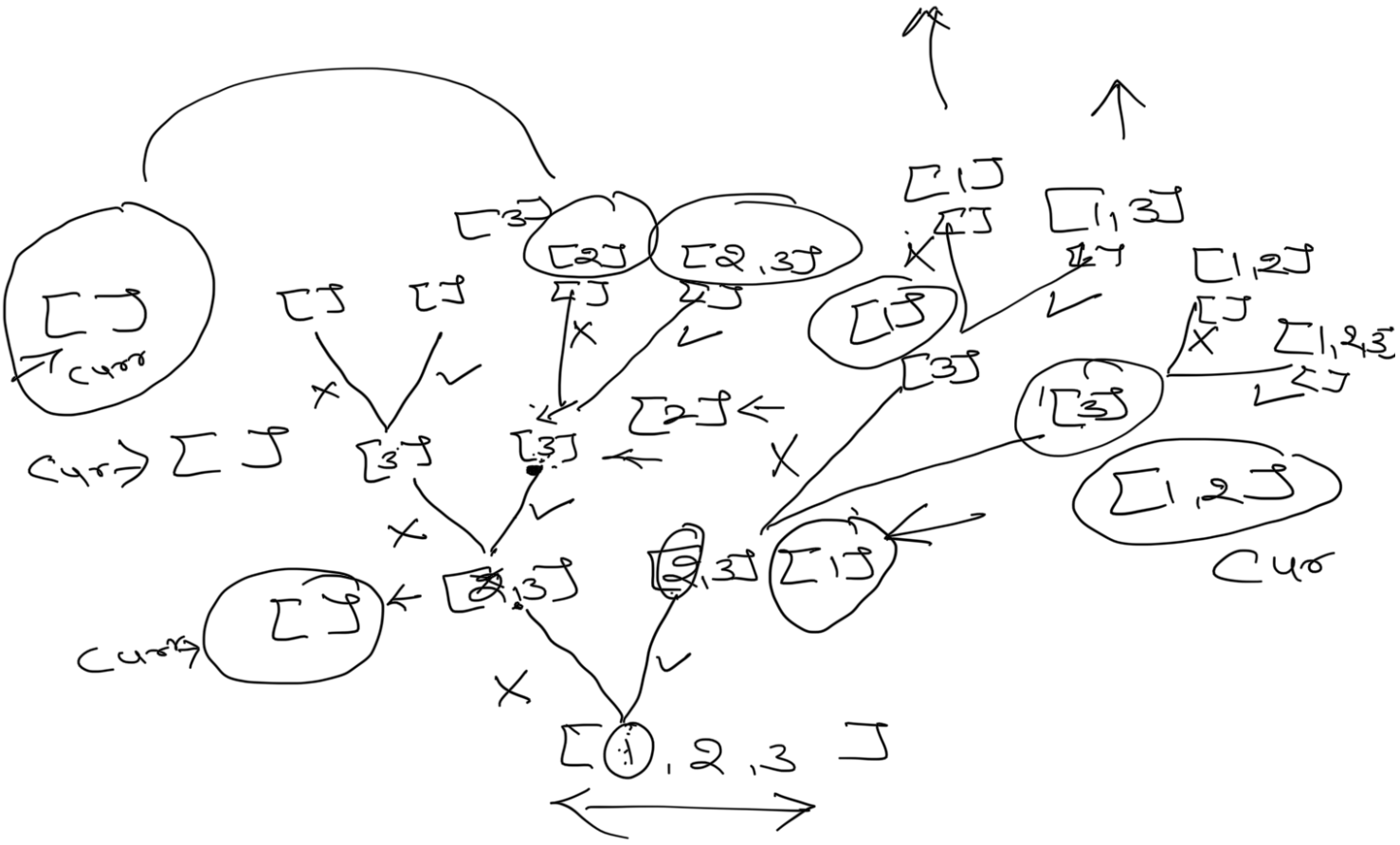
3 ⇒ 2^3 ⇒ 8

[ ] ⇒ [ ]

0 ⇒ 2^0 ⇒ 1

⇒ [1, 10, 20, 30, 40]

⇒ 2^5 ⇒ 32

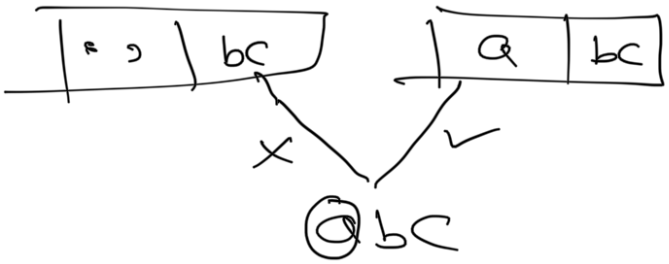


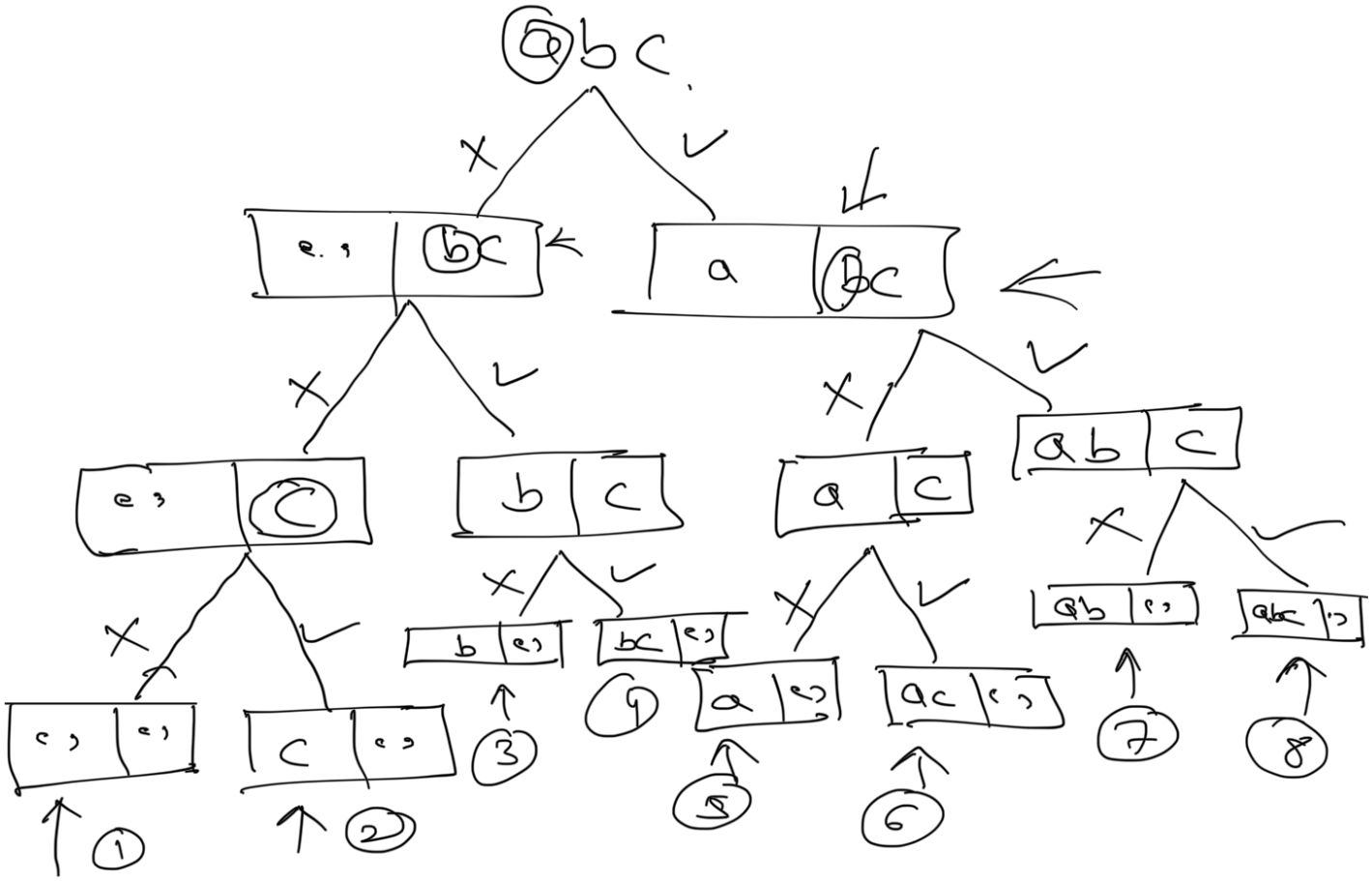
Output ⇒ [[ ], [3], [2], [2,3]

[1], [1,3], [1,2]

[1,2,3]]

「②、3」


bc
a
bc
x
v
@bc

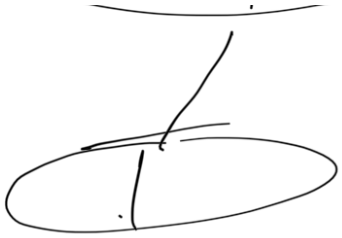

@bc
x
v
bc
a
x
v
x
v
b
c
a
c
ab
c
x
v
x
v
x
v
b
bc
a
ac
ab
abc
3
4
5
6
7
8
1
2

Output => ['', 'c', 'a', 'ac', 'b', 'bc', 'ab', 'abc']

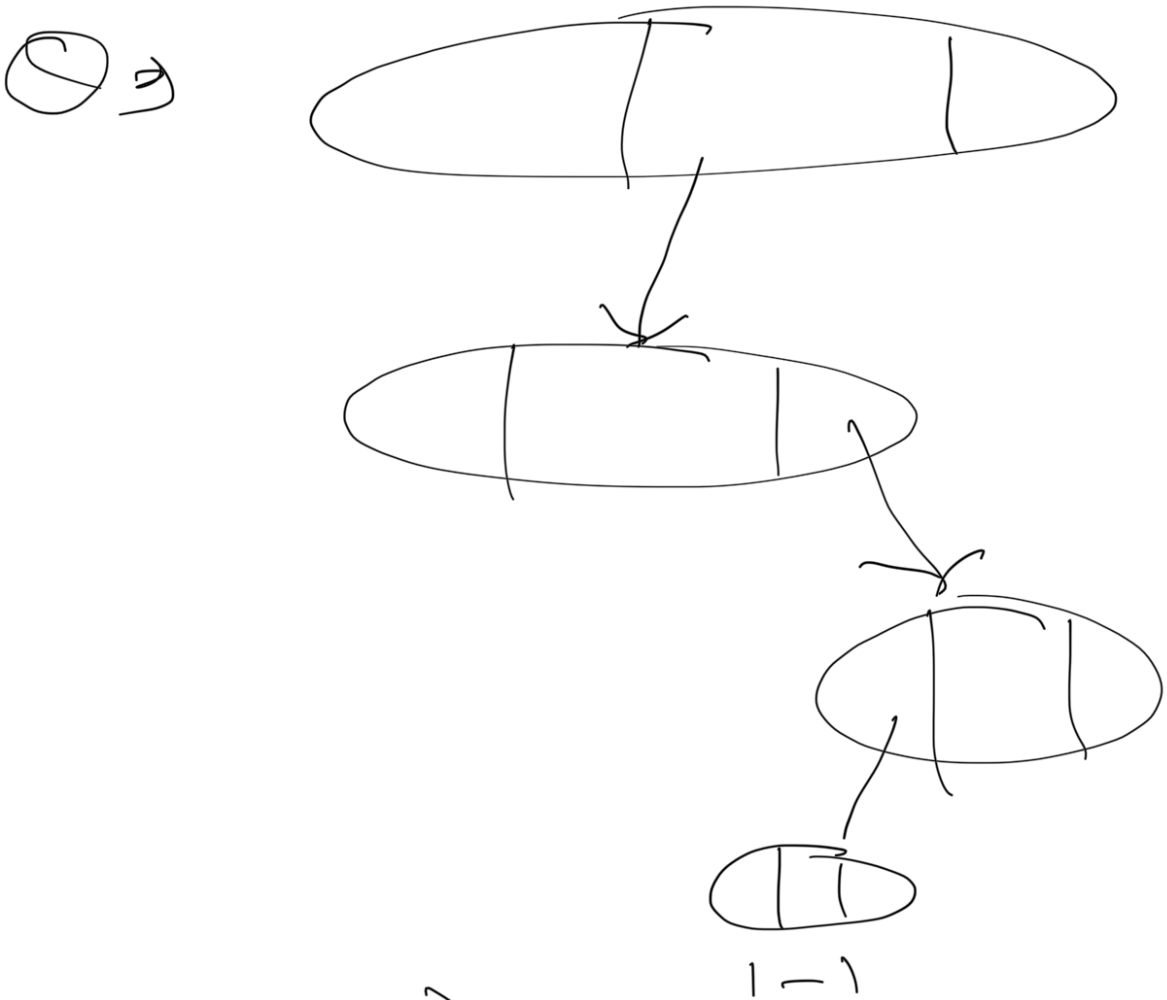
2^n => T.C (2^n)

=> => T.C (3^n) <=

=> S.C => O(n)

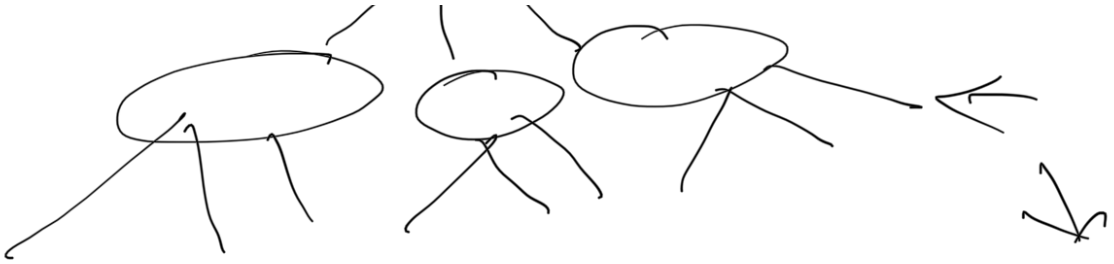


1 1 → $\log_2 N$ ←



1 - 1

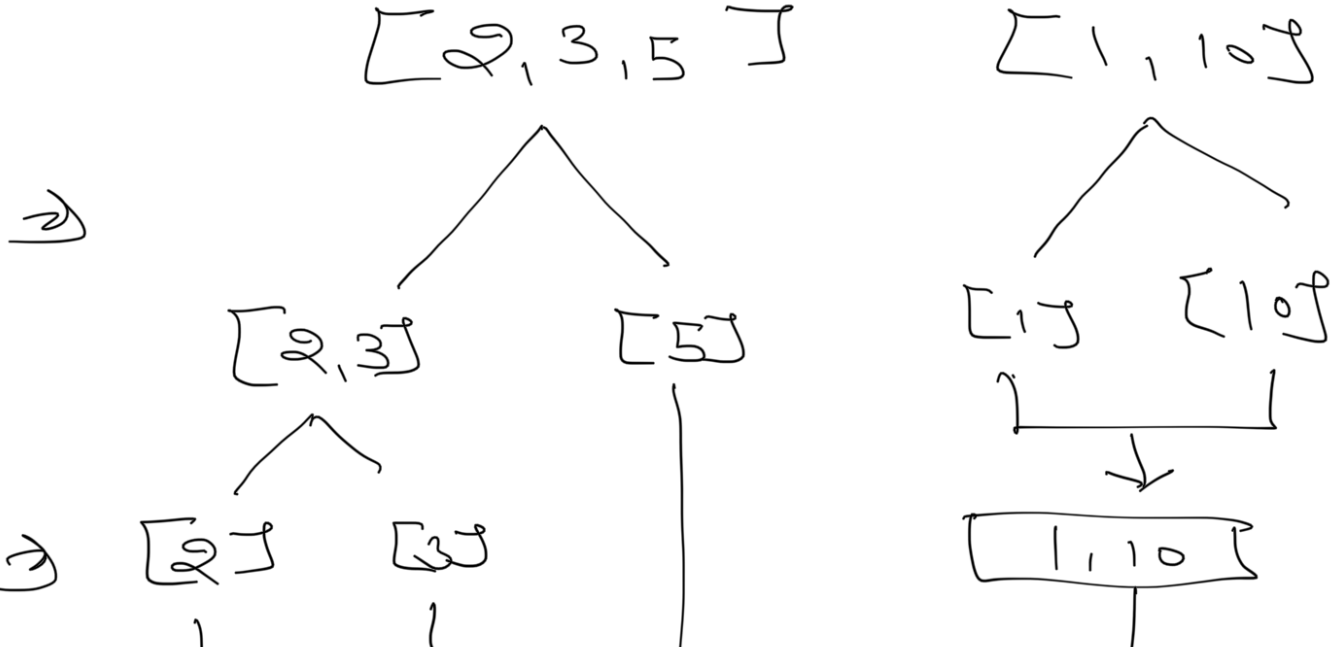
→ T.C: $\log_3 N$

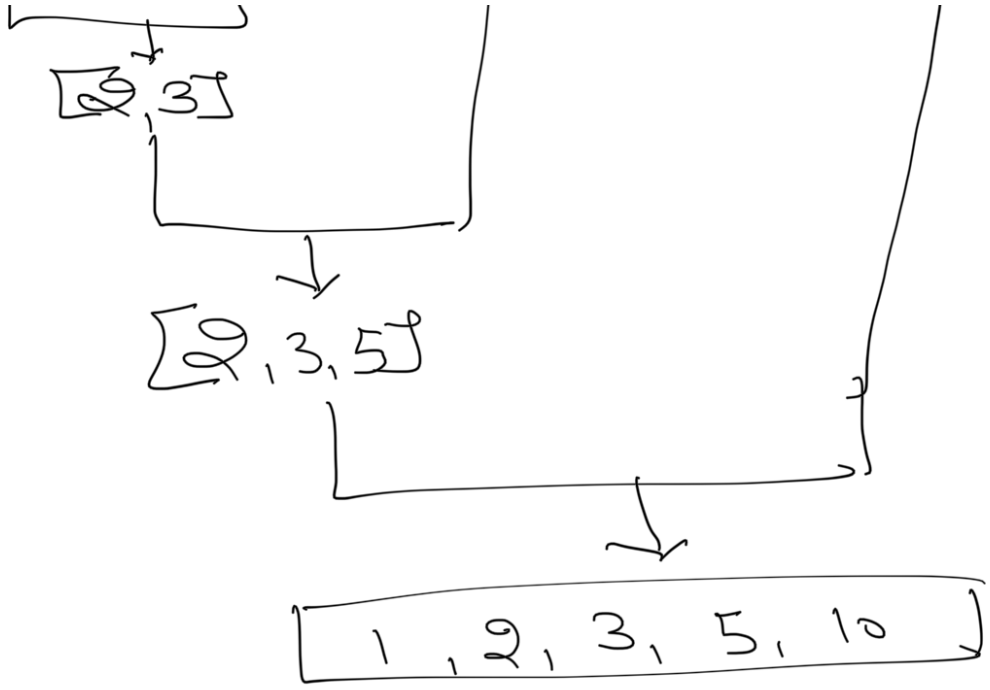


T.C 3^n

2^n

[2, 3, 5, 1, 10]





2) Home-work

a) Write an Algo to merge two sorted Arrays

Input a) [2,3,5] b) [1,10]

o/p [1,2,3,5,10]