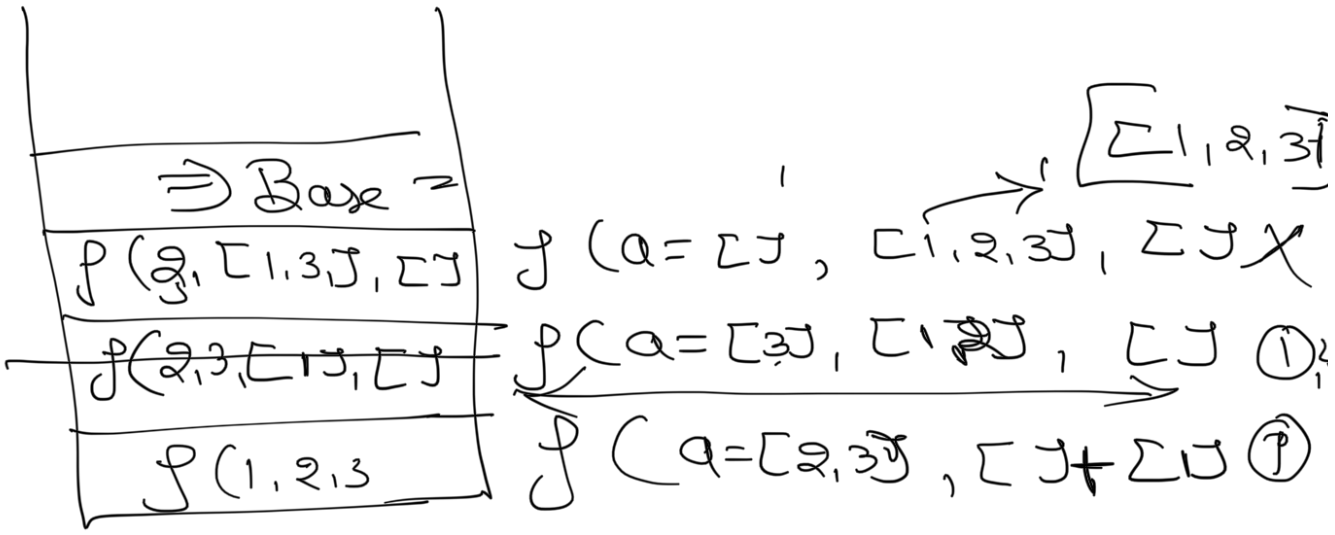
Recursion

[

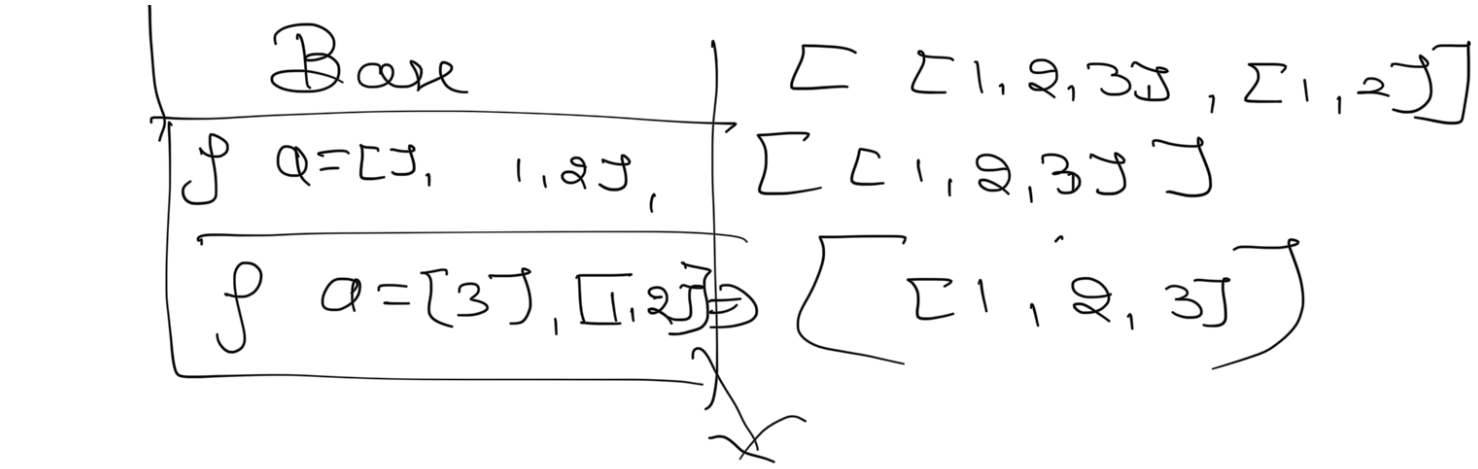
① ⇒ get_subset (arr [1:], ⇒ curr_sub + [arr[0]], output)

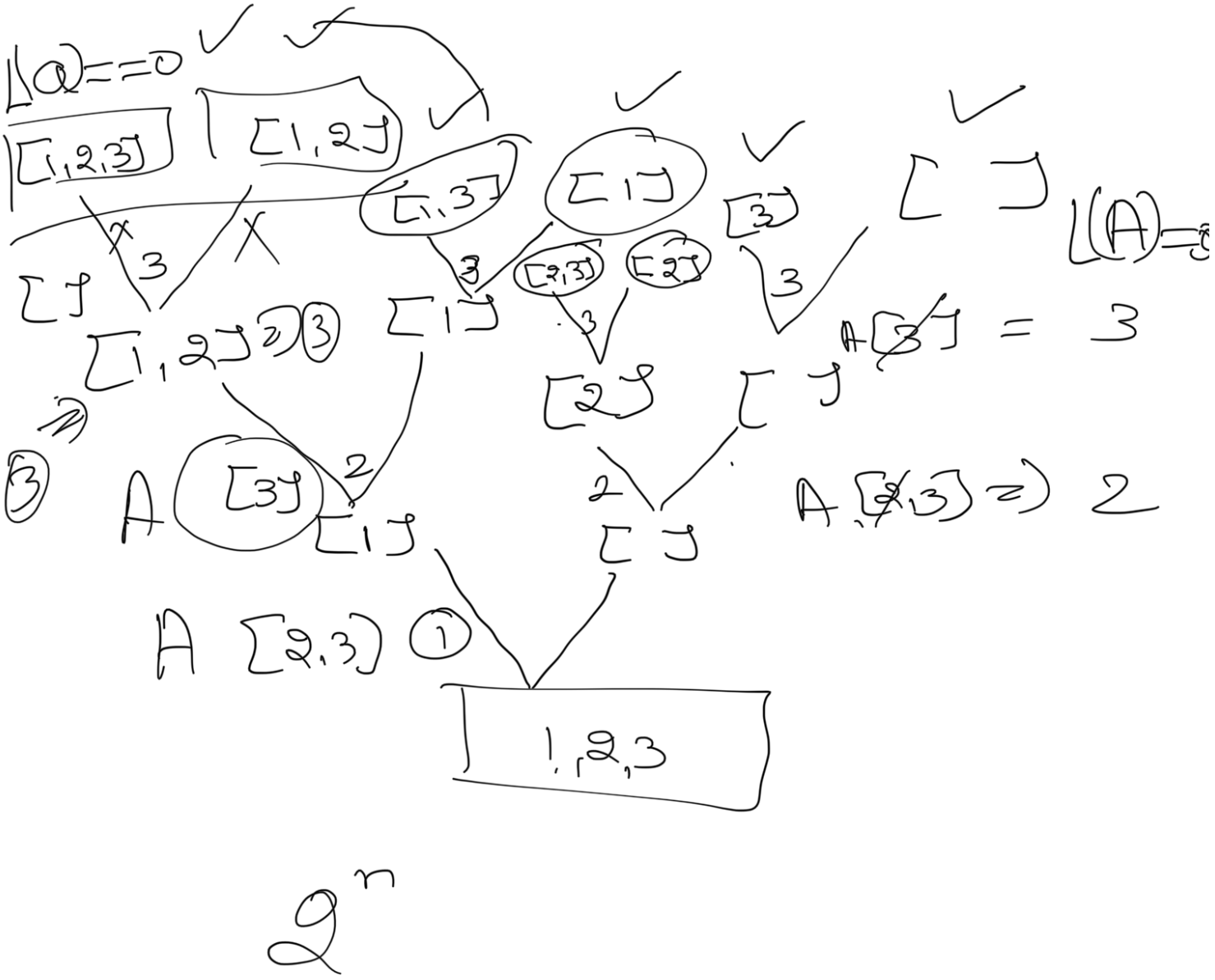
② ⇒ get_subset (arr [1:], curr_sub, output)

arr = [1, 2, 3] ← arr[0]



[ ],

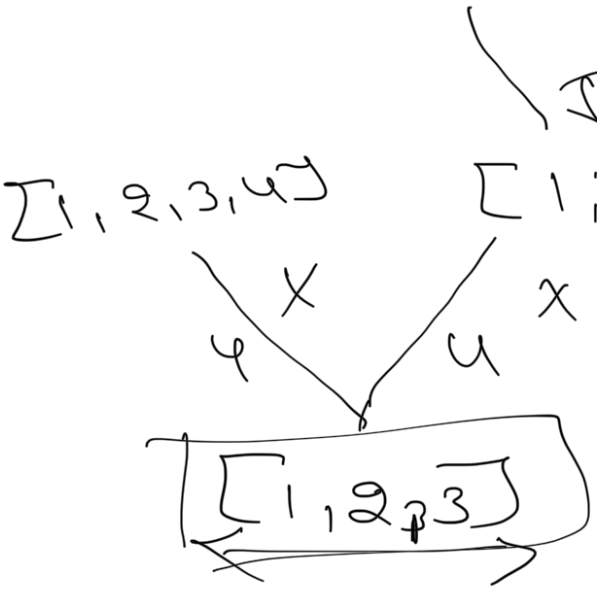




2^n

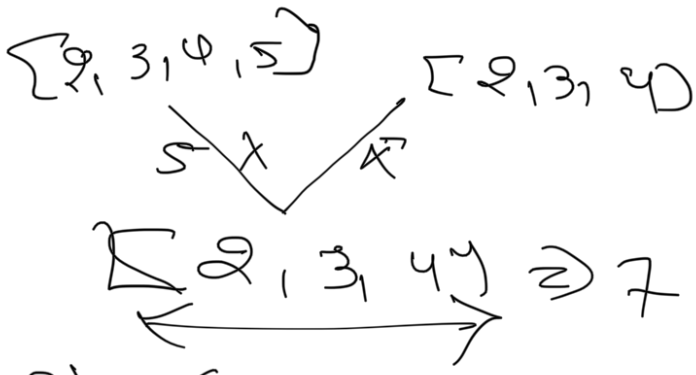
Dynamic Programming

- fibonacci
- Target Sum Sub
  - with rep
  - without repitition
- Subset [1,2,3] ?



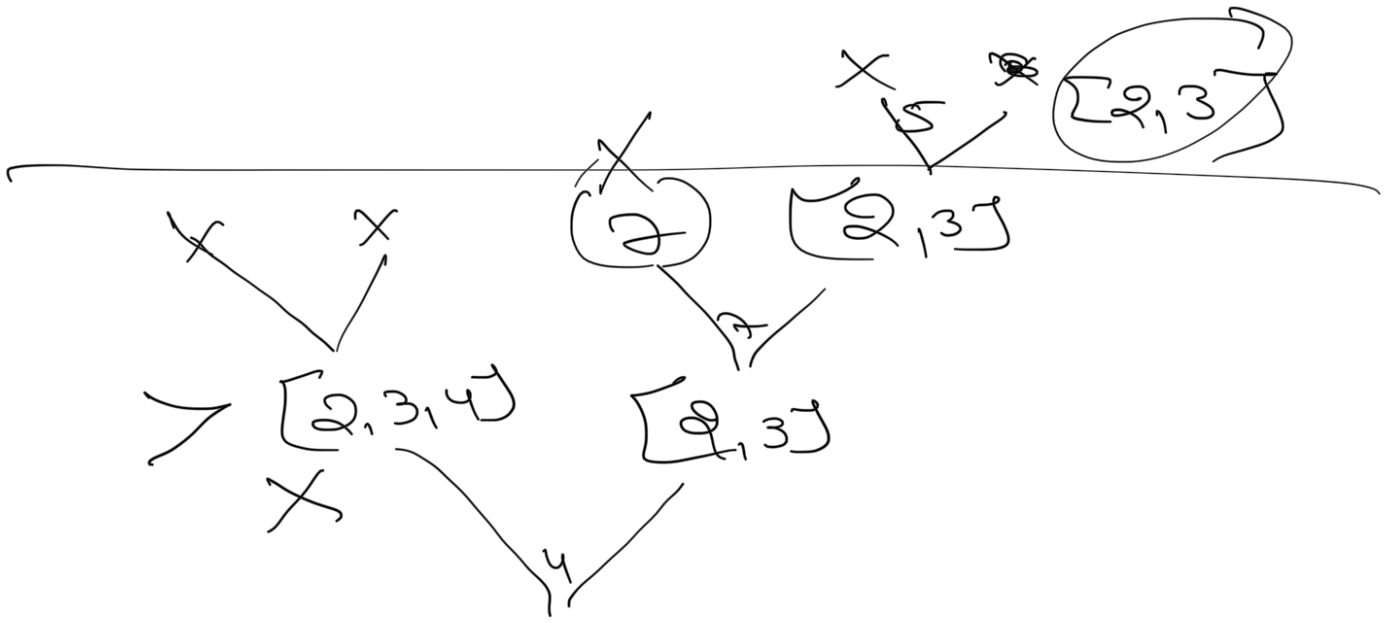
Target = 6

[4,5]

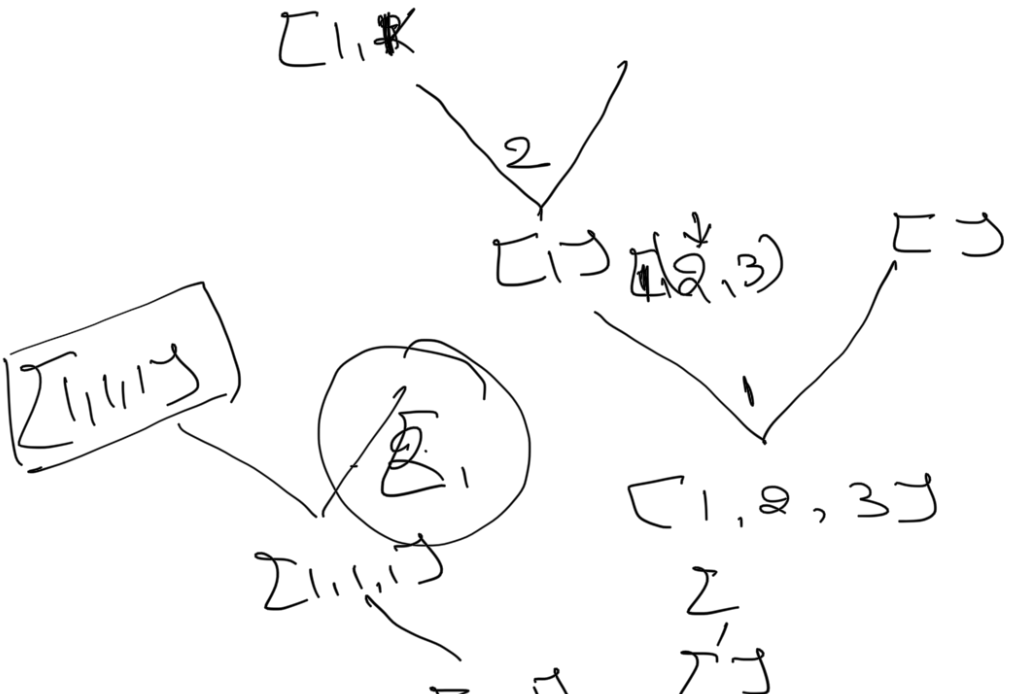


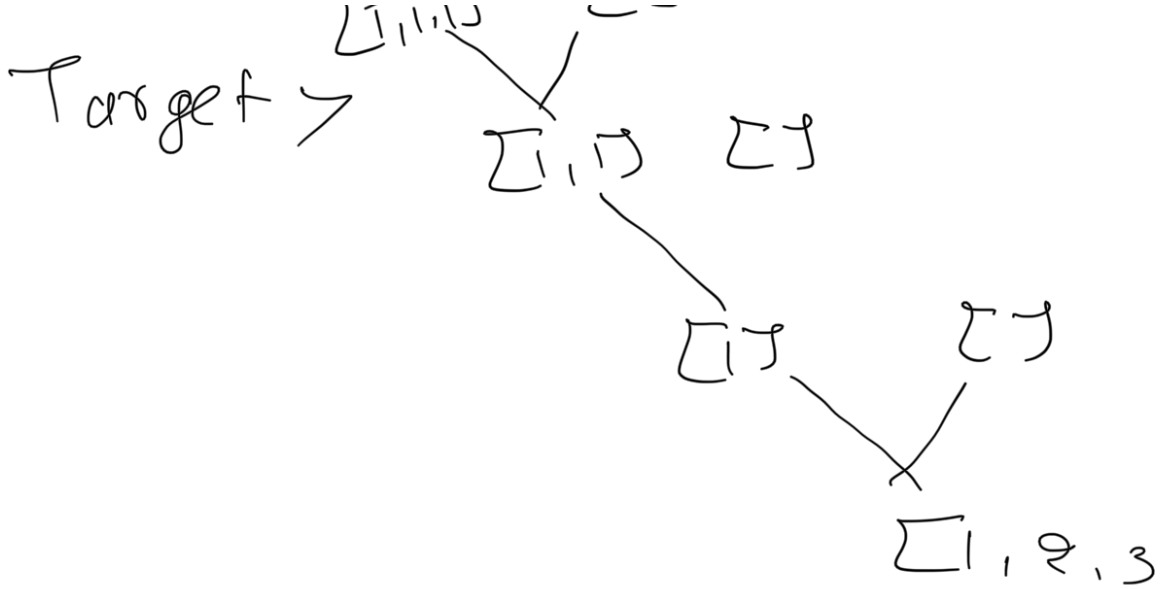
Target = 6

[2,3],4 =



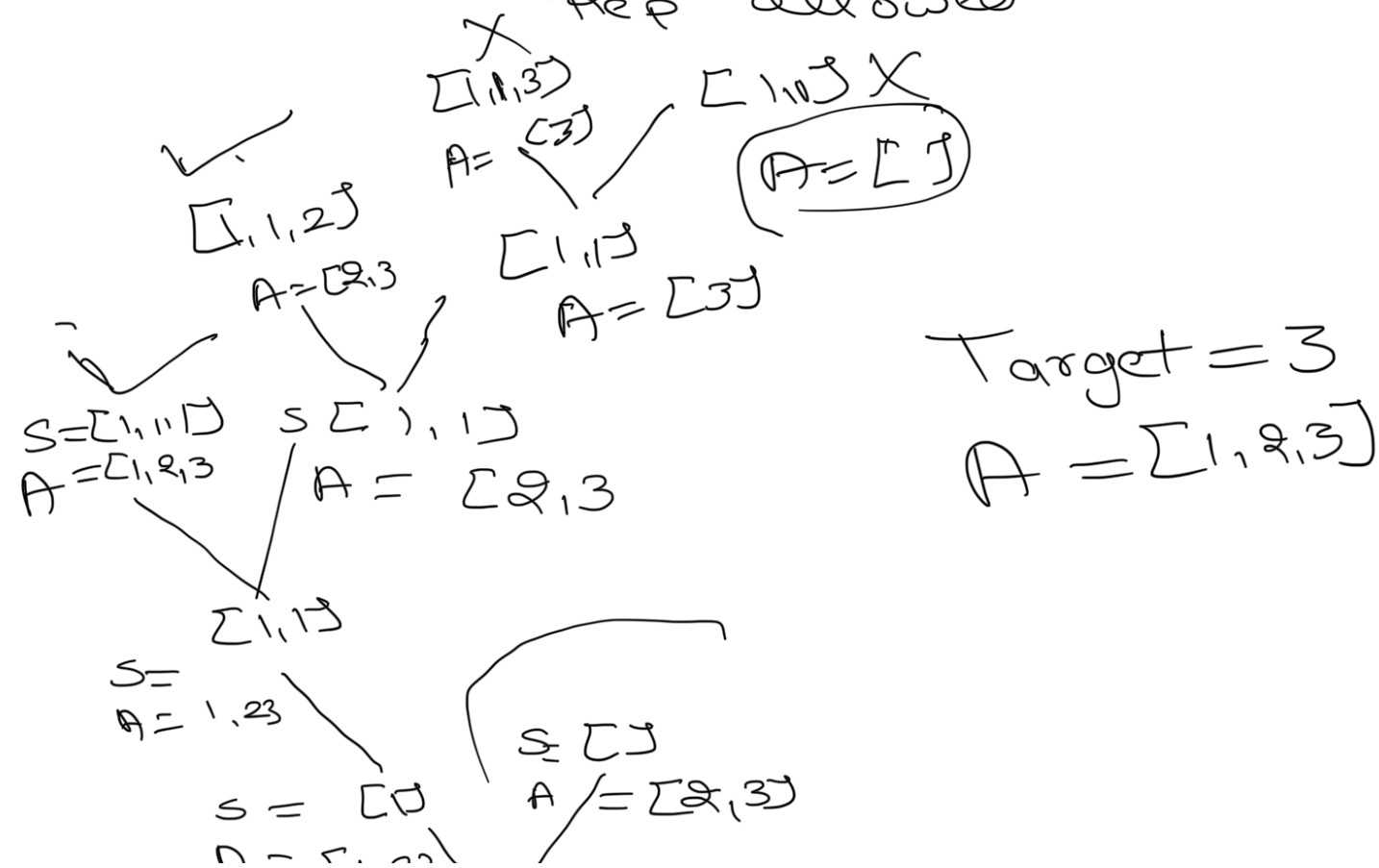
Target = 6





Home-work : Draw call-Tree

Rep allowed



サー {1,2,3}

{1,2,3}