Approach.

1. Recursion
2. Memorization
3. Tabulation

} to get better understanding and grasp of problem

Partition in to subset.

n = no. of element

k = no. of subset

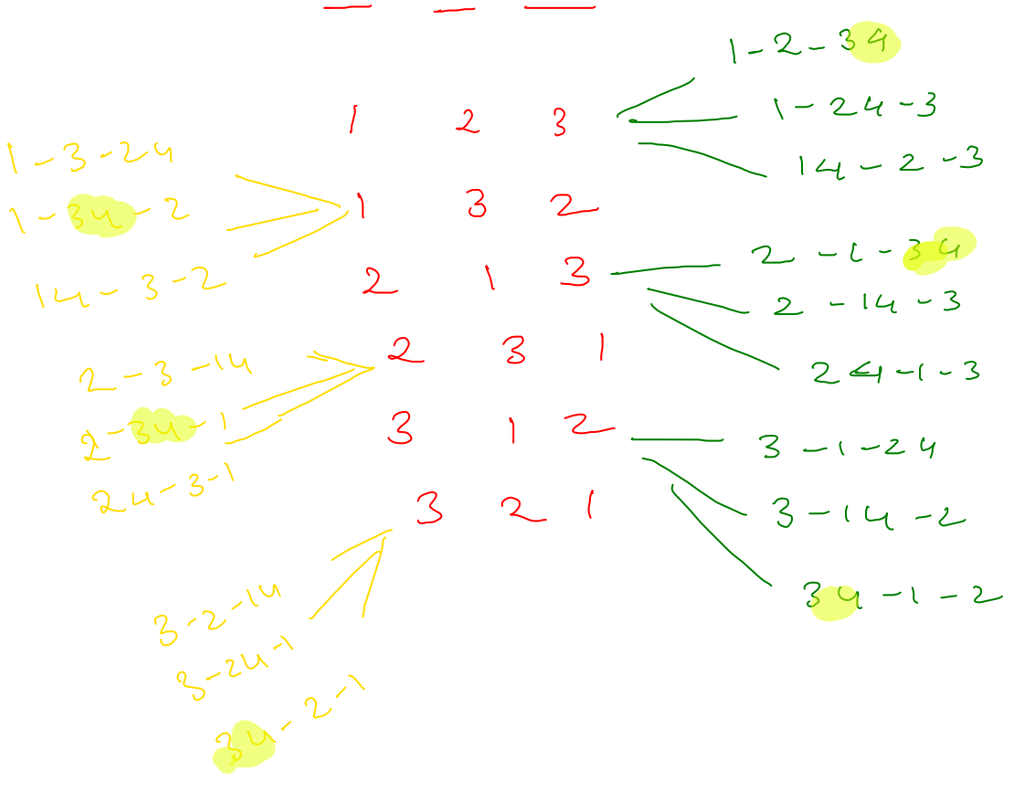
print — no. of ways in which elements can be partitioned in k non empty subset

n = 4 elements (1, 2, 3, 4)

k = 3

1 2

— — —



Total permutation

1 2 3 = 6 permutation.

with each 2 can put left element in 4 sets.

∴ 6 × 3 = 18 permutation.

we need combinations

n k

4 - 3

3 - 3

3 - 4

1234

n ? k - - -

n = k - 1 way 1 2 3

n < k - 0 way

1 2 3 ⊘ empty

1 - 23 - 4

13 - 2 - 4

12 - 3 - 4

n c k = 4 C 3 or 5 C 3

1 2 3 1 2 3 4 5

How to come up with solution

1 2 3 4 5

_ _ _

↑
space.

only 3 elements you can give space.

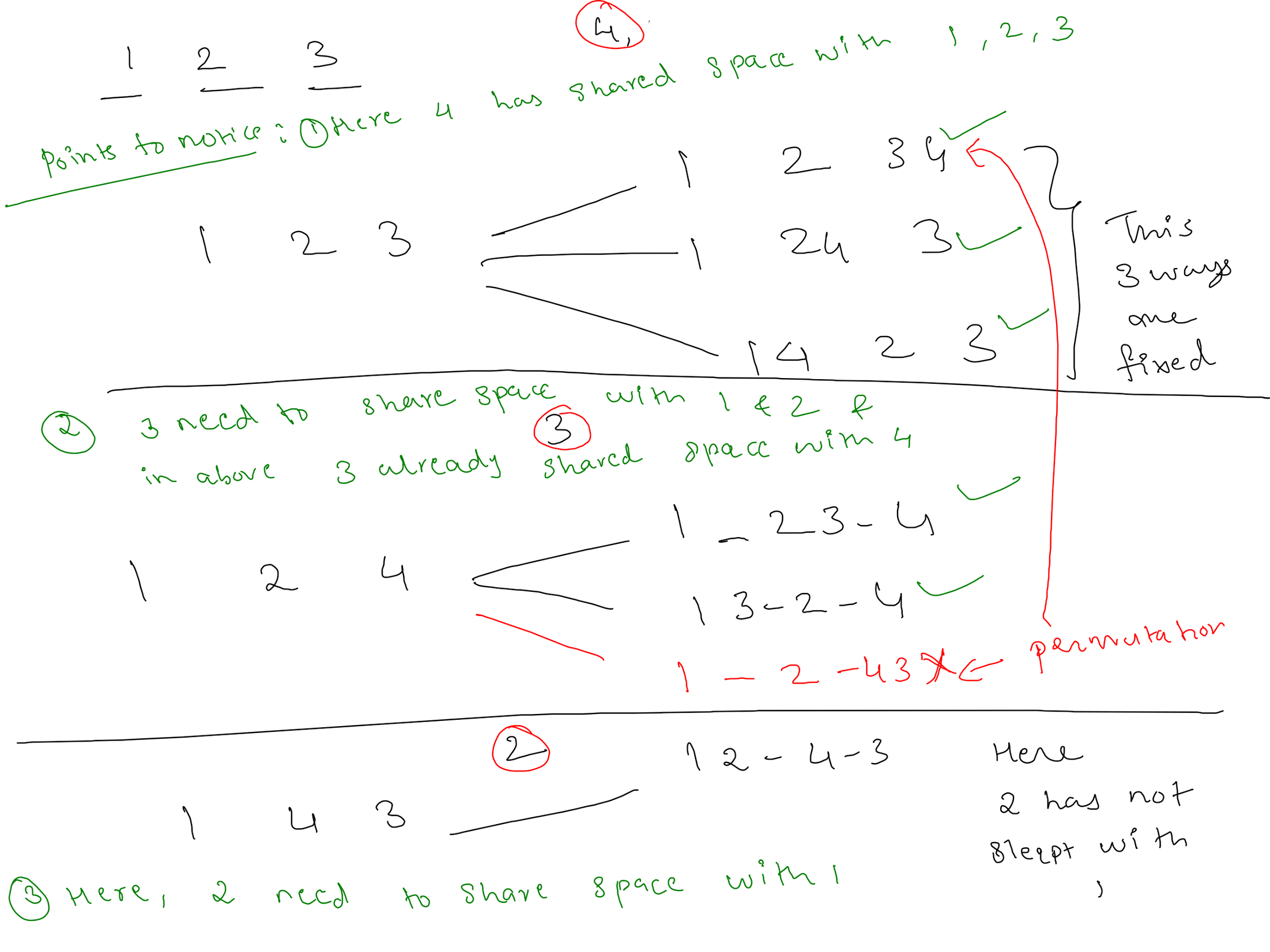
remaining (5-3) = 2 element need to share space.

Let say if I put 1, 2, 3 occupies space

1 2 3

_ _ _

other 2 element (4, 5) need to be shared with 1, 2, 3


1 2 3
(4)
Points to notice : (1) Here 4 has shared space with 1, 2, 3
1 2 3
1 2 3 4
1 2 4 3
1 4 2 3
This 3 ways one fixed
(2) 3 need to share space with 1 & 2 &
in above 3 already shared space with 4
(3)
1 2 4
1 _ 2 3 - 4
1 3 - 2 - 4
1 - 2 - 4 3
permutation
(2)
1 4 3
1 2 - 4 - 3
Here 2 has not slept with 3
(3) Here, 2 need to share space with 1

total 6 ways :

when

$n = 4$

$k = 3$

④

1 2 3 — 1 – 2 – 34

1 – 24 – 3

14 – 2 – 3

③

1 2 4 — 1 — 23 – 4

13 – 2 – 4

② 6 ways

1 4 3 — 12 – 4 – 3

Try to divide problem in subproblem. HOW ?

Analogy → 4 log hai & 3 ghar hai.

Subproblem can be

3 log 3 ghar

3 log – 2 ghar

3 log – 1 ghar

3 log – 0 ghar

$n = 4$

$k = 3$

Looking at tree, assumptions:

(3,3) – muje 3 ka subset dega. from 3 element aur usme muje 1 last element ko har ek k sath share karna padega.

(3,2) – muje 2 ka subset dega. from 3elements aur jo last element rahega use append karke 3 ka subset bana dunga.

(3,1) – muje 1 ka subset dega from 3 elements.

means 123 in 1 set

and last element ko 1 set me daale toh 1 empty set bachega

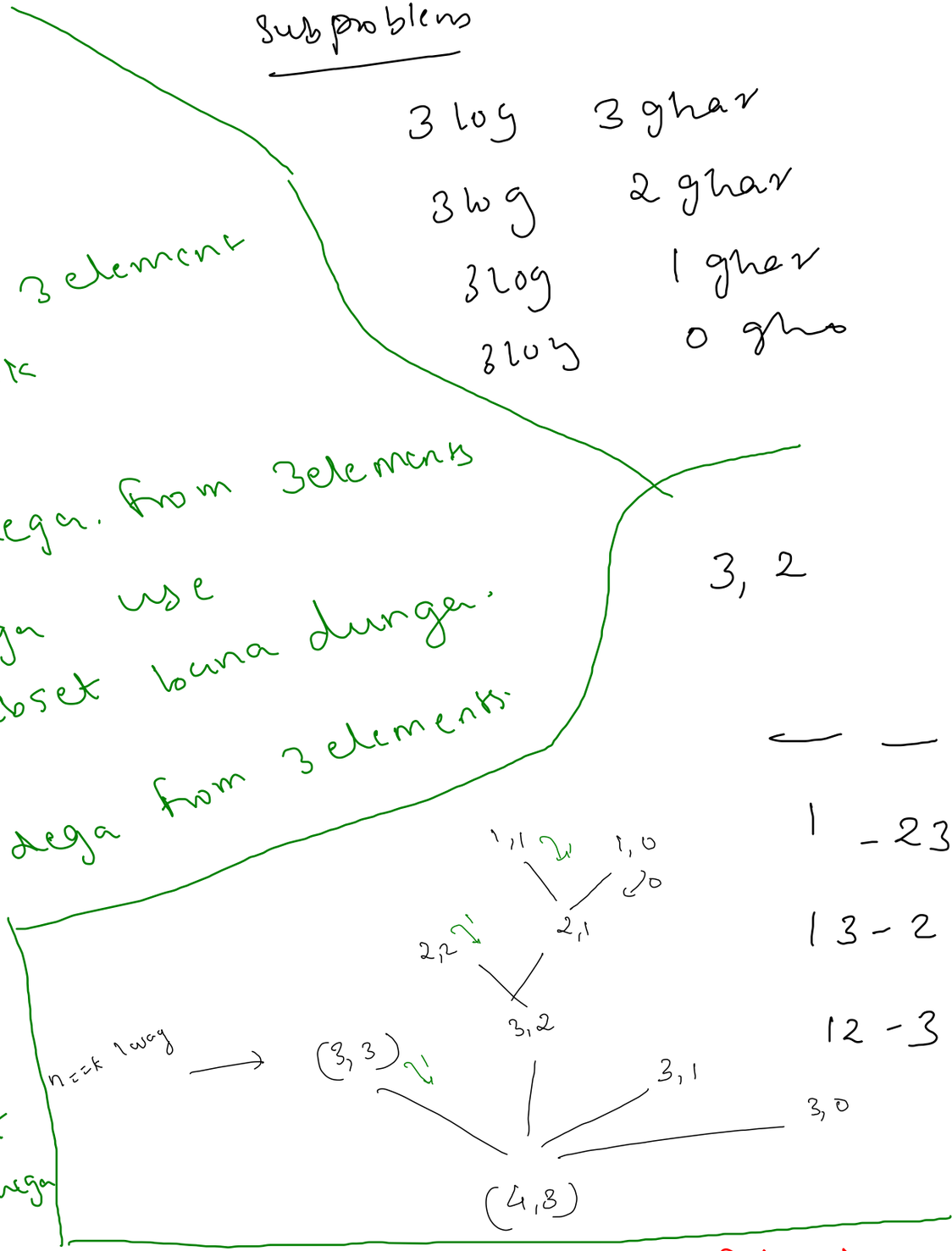
Subproblems

3 log 3 ghar

3 log 2 ghar

3 log 1 ghar

3 log 0 ghar



3, 2

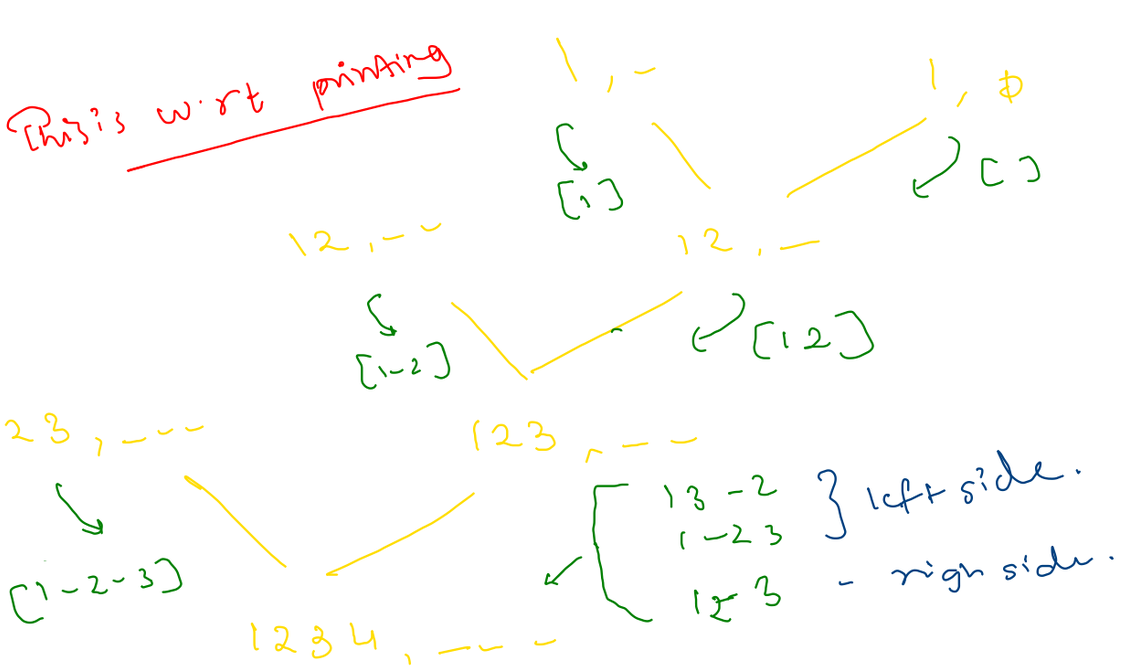
1 – 23

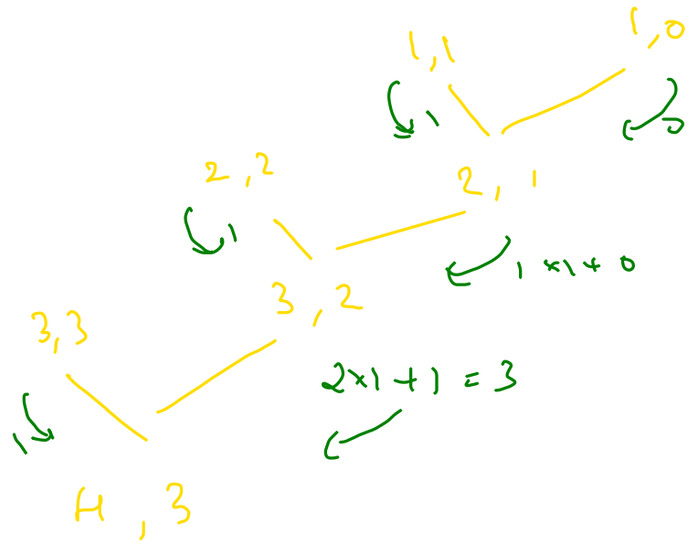
13 – 2

12 – 3

How? 123 4 — So we will not consider 3,1 as option in Euler.

This is wrt printing





on left side, I will iterate 4 with each subset.

i.e. 1-2-34, 1-24-3, 14-2-3

from right side, I get 2 ka subset. So I will append

4 or each ans

13-2-4, 1-23-4, 12-3-4

$$f(n,k) = k \times f(n-1,k) + f(n-1,k-1)$$

```java
// ~~~~~~~~~~~~~~~~Partition into K Subset~~~~~~~~~
public static long partitionKSubset(int n, int k) {
    long[][] dp = new long[n + 1][k + 1];

    for(int i = 0; i <= n; i++) {
        for(int j = 0; j <= i && j <= k; j++) {
            if(j == 0) {
                dp[i][j] = 0;
            } else if(i == j) {
                dp[i][j] = 1;
            } else {
                dp[i][j] = dp[i - 1][j - 1] + j * dp[i - 1][j];
            }
        }
    }
    return dp[n][k];
}
```

n ≥ k solve

n = k return 1

n < k return 0

n = 4

k = 3

→ k space

n \ k	0	1	2	3
0	0	0	0	0
1	0	1	0	0
2	0	1	1	0
3	0	1	1×2+1 = 3	1
4	0	1	3×2+1 = 7	1×3+3 = 6

n

log n

dp[n][k] = dp[n-1][k] * k

+ dp[n-1][k-1]