Pre-requisite →

Patterns

1) fib

2) recurrence, state

3) LIS, LCS, Rod Cutting, Min Coin change

4) Knapsack — 0-1, Subset Sum

5) Catalan No

Alphacode (Spoj) → fib

25114 → YKD

→ BEAAD

25114

25114

↳ Brute force → We can try all possible comb.

1 — 26

2 5 1 1 4

for every digit we have 2 choices either consider it alone or consider it with a digit adjacent to it.

$f(s, i)$ = $\begin{cases} f(s, i-1) \\ + \\ f(s, i-2) \end{cases}$ → if $(s[i-1]*10 + s[i] \leq 26)$

The no. of ways in which we can decode the string s from index (0 - i)

s[0 - i]

2 5 1 1 4

i-1

i

State depends on 1 parameter. → 1D dp

25114

$\underbrace{251}_{x}$ $\underline{14}$

$\underbrace{2511}_{y}$ $\underline{4}$

114

2

1 $\underline{14}$

11 $\underline{4}$

$\boxed{\text{AN}}$ + $\boxed{\text{AAD KD}}$

Unique Binary Search Trees

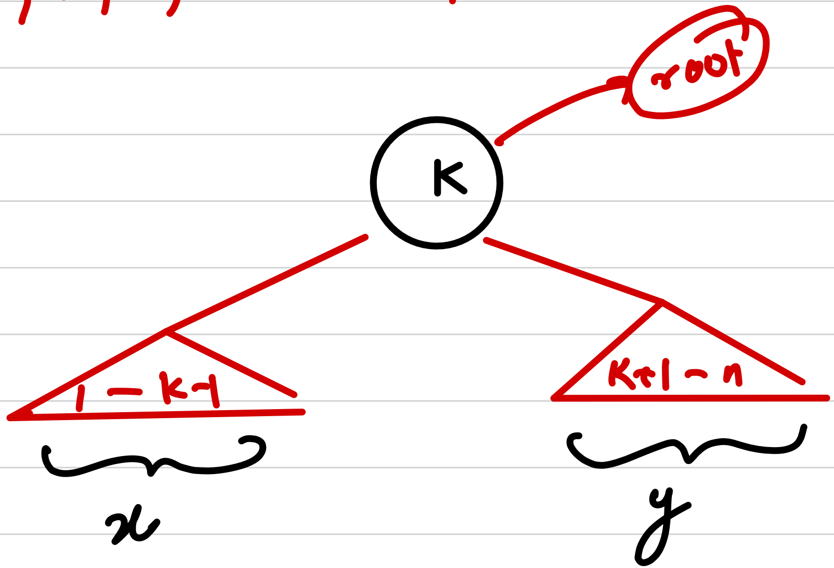
$n \rightarrow$ no. of nodes

$f(n) \rightarrow$ this function will calculate the no. of structurally unique BST's

every node from $1-n$ can become root once.

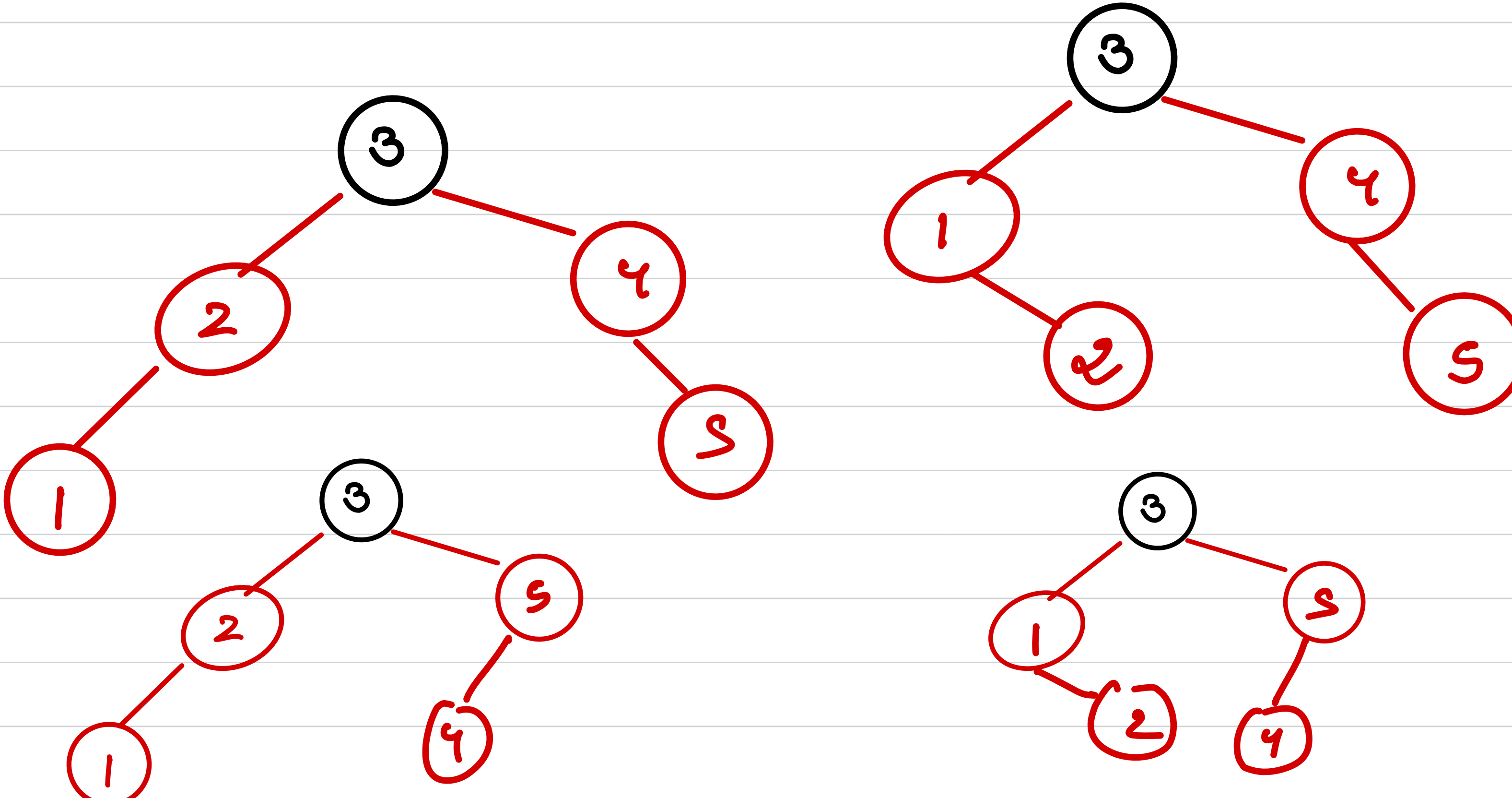
K

1, 2, 3, ... n



$\longrightarrow x \times y$

1, 2, 3, 4, 5



if LST can be formed in x ways and RST can be formed in y ways then for a given root no. of trees possible is

$$\boxed{x \times y}$$

$$g(i) = g(i-1) \times g(n-i)$$

$g(i)$ → no. of tree we can create keeping i as the root node

$g(i-1)$ → no. of ways of forming LST

$g(n-i)$ → no. of ways of forming RST

$$f(n) = G(1) + G(2) + G(3) \ldots\ldots G(n)$$

$$\boxed{f(n) = \sum_{K=1}^{n} G(K)}$$

$G[0] = 1$
$G[1] = 1$

→ for calculating any value $\underline{\underline{i}}$ we need already precomputed value of $\underline{\underline{(1 - i{-}1)}}$

→ $f(n)$ → no. of ways to form unique BST

$f(i) = f(i-1) \times f(n-i)$

ans → $\underline{\underline{\Sigma f(i)}}$