



DP on trees

Dev Karan Singh (devkaran1231)
Expert at codeforces (1817)
5 star at codechef (2041)



DP on Trees

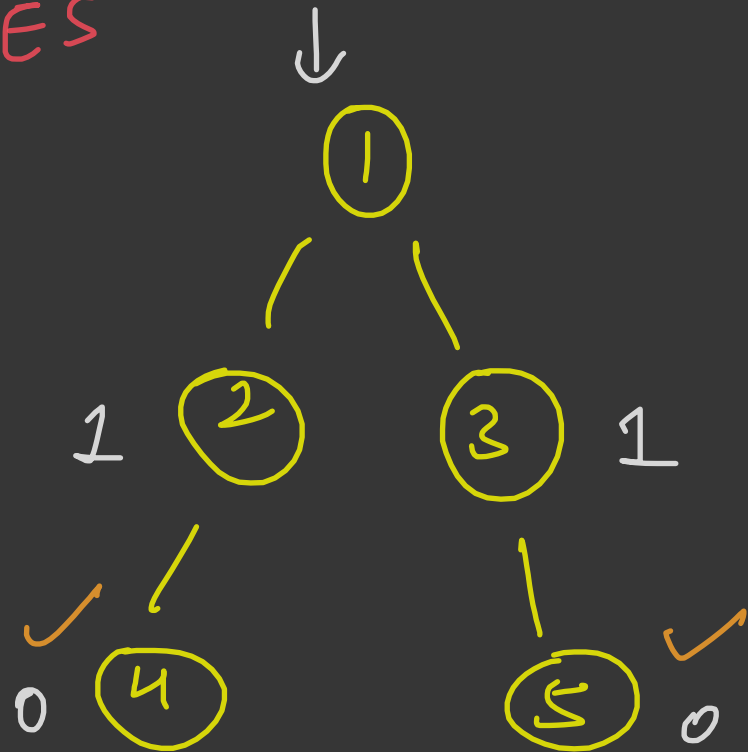
- The simplest DP on Trees Problem -> [Subordinates](#)
- General DP on trees -> [Tree Matching](#)
- Re-rooting DP (UBER special)
 - [Tree Distances 2](#)
 - [Tree with Max Cost](#)

DP on trees →

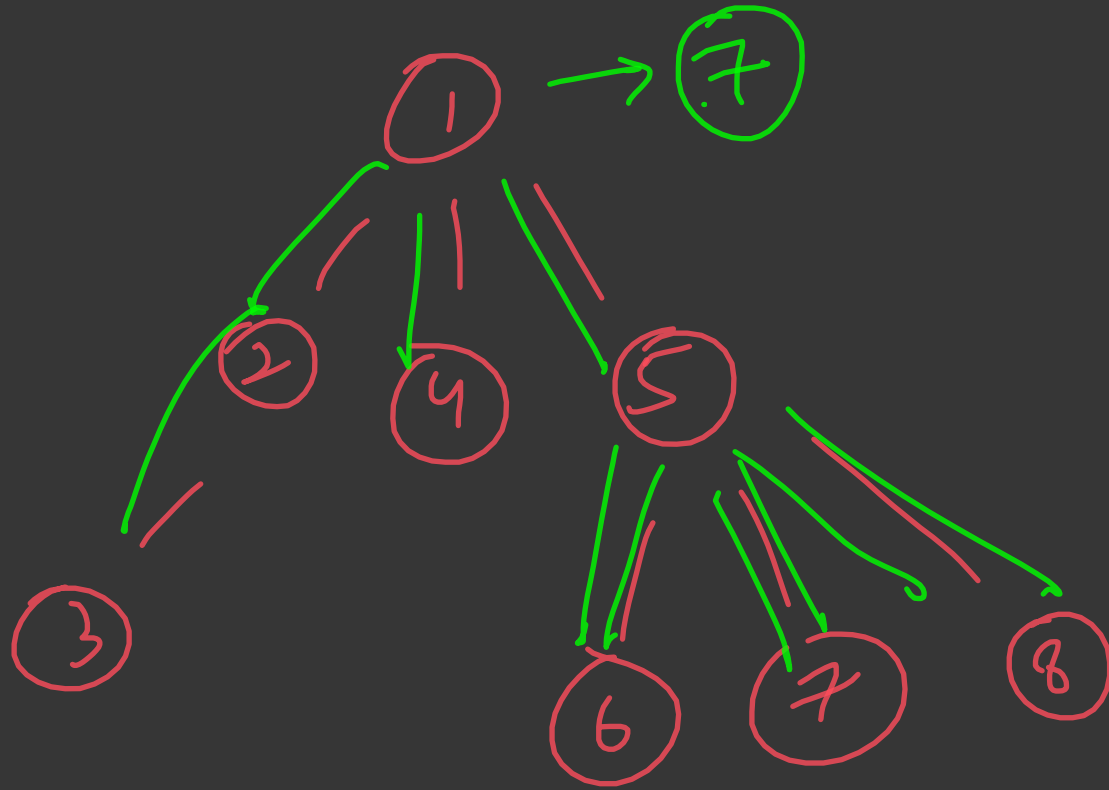
Q → subordinate CSES

1 2 3 4 5
→ 4 1 1 0 0

$$dp[1] = \cancel{dp[2]} + \cancel{dp[3]} + 2$$



eg.



{

for (auto it : adj[node]) {

0	0	0	0	0	0	0	0
1	2	3	4	5	6	7	8

if (it == par) continue;

dfs(~~it~~^{2, 4, 5}, node, adj),
dp[node] += dp[it];

}

dp[node] +=

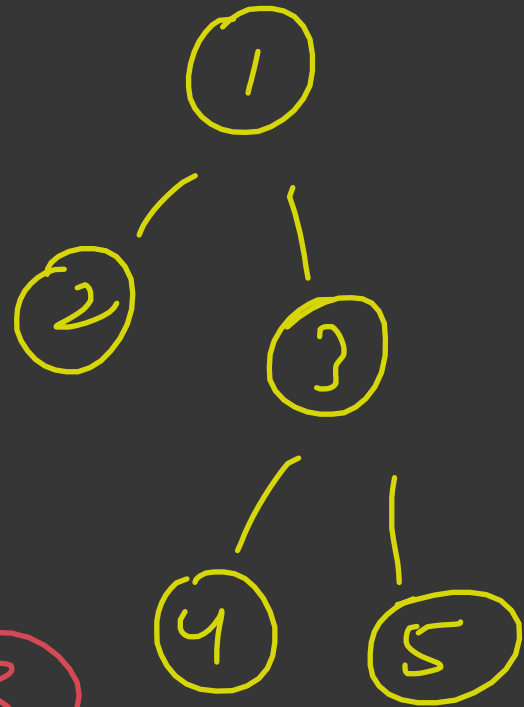
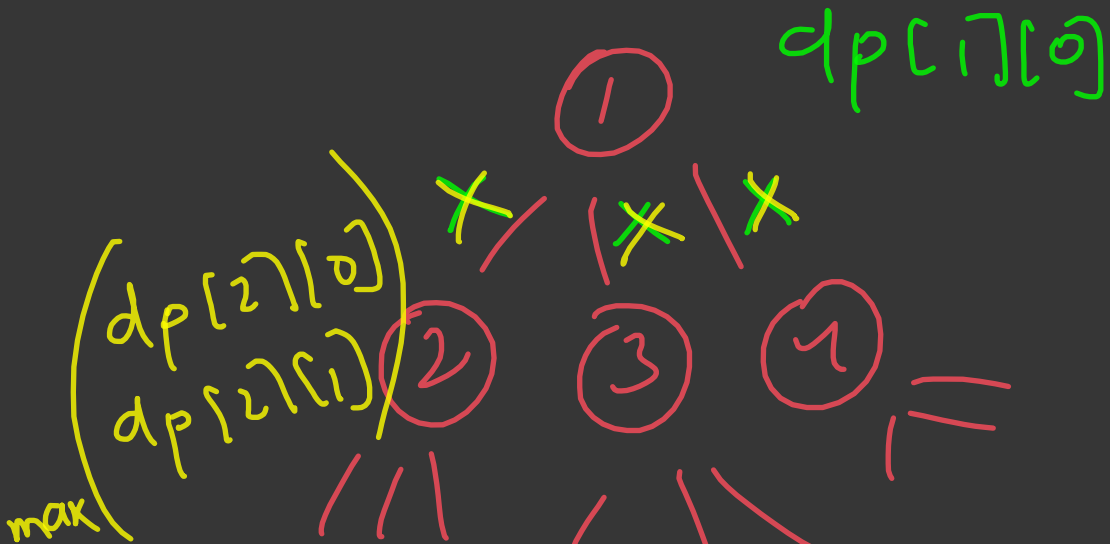
adj[node].size() -

(node == 1 ? 0 : 1);

}

no of
children
of node

Q2 Tree Matching



$$\max(dp[1][0], dp[3][1])$$

$dp(i, 0) \Rightarrow$ ans for subtree rooted at i such that there is no edge between i and any of its child.

$dp(i, 1) \Rightarrow$ ans for subtree rooted at i such that there is an edge between i and only one of its child.

2D dp

$dp[i][flag]$

$dp[i][0]$

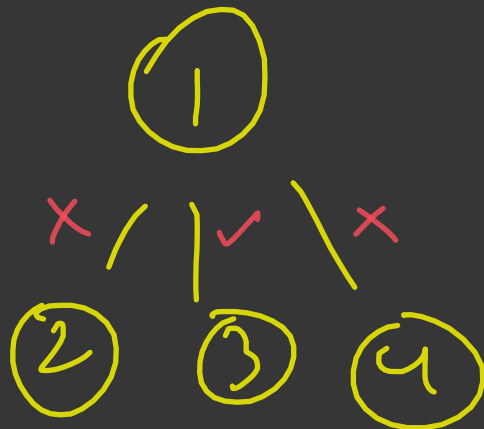
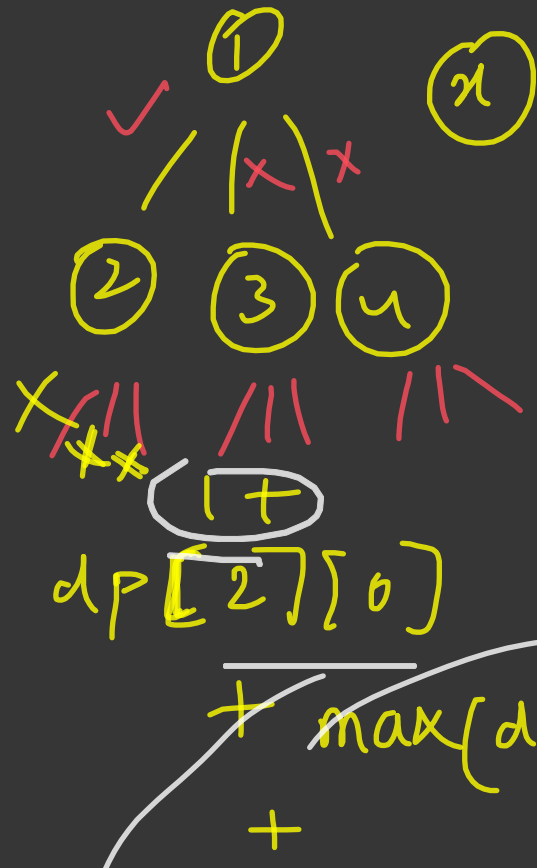
$dp[i][1]$

Base case ?

$dp[leaf][0] = dp(leaf)$
 $[1] = 0$

$$\text{dp}[node][0] += \max(dp[it][0], dp[it][1]);$$

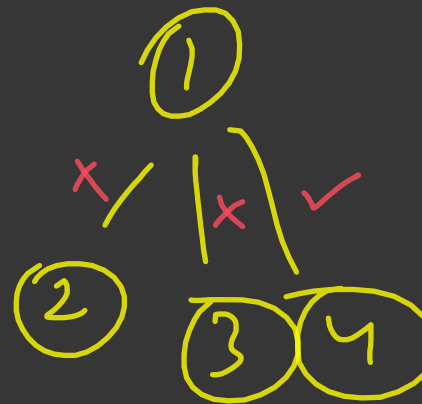
$$dp[node][1] =$$



$$1 + dp[3][0]$$

$$+ \max(dp[2][0], dp[2][1])$$

$$+ \max(dp[4][0], dp[4][1])$$



$$1 + dp[4][0]$$

$$+ \max(dp[2][0], dp[2][1])$$

$$+ \max(dp[3][0], dp[3][1])$$

$\max(dp[4][0], dp[4][1]);$ ——— (✓)

→ 2

$dp[1][0] = \max(dp[2][0], dp[2][1])$

$+ \max((3,0), (3,1))$

$+$

$\max((4,0), (4,1))$

→ 2

$$\text{Z} \rightarrow = dp[1][0] -$$

$$\max(dp[2][0], dp[2][1]),$$

$$dp[1][1] = \max(dp[1][1],$$

$$\underline{1 + dp[2][0] + dp[1][0]})$$

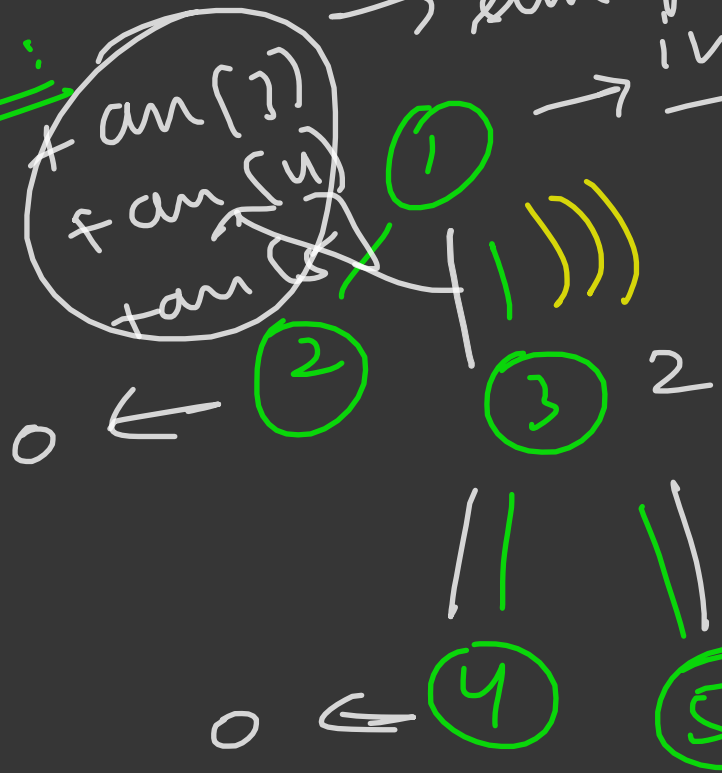
if selecting it
from node

✓ — $\max(dp[2][0],$
 $\textcircled{it} \quad dp[2][1])$
 \textcircled{node}

$$dp[node][1] = \max(dp[node][1], 1 + dp[it][0] + dp[node][0] - \max(dp[it][0], dp[it][1]));$$

Re Rooting DP \rightarrow

eg:



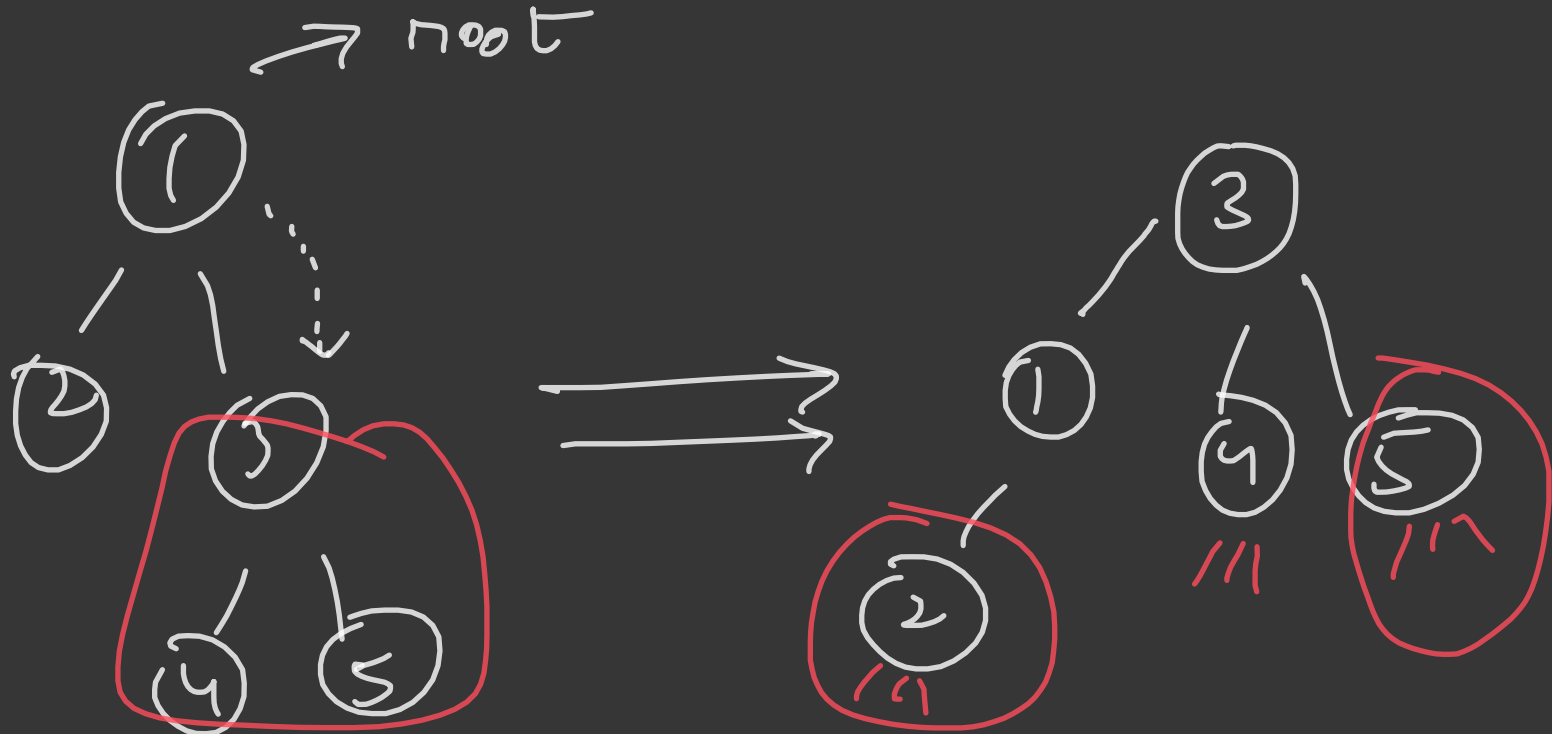
\rightarrow sum of values of nodes is subtree(1)
 $\rightarrow in[1]$

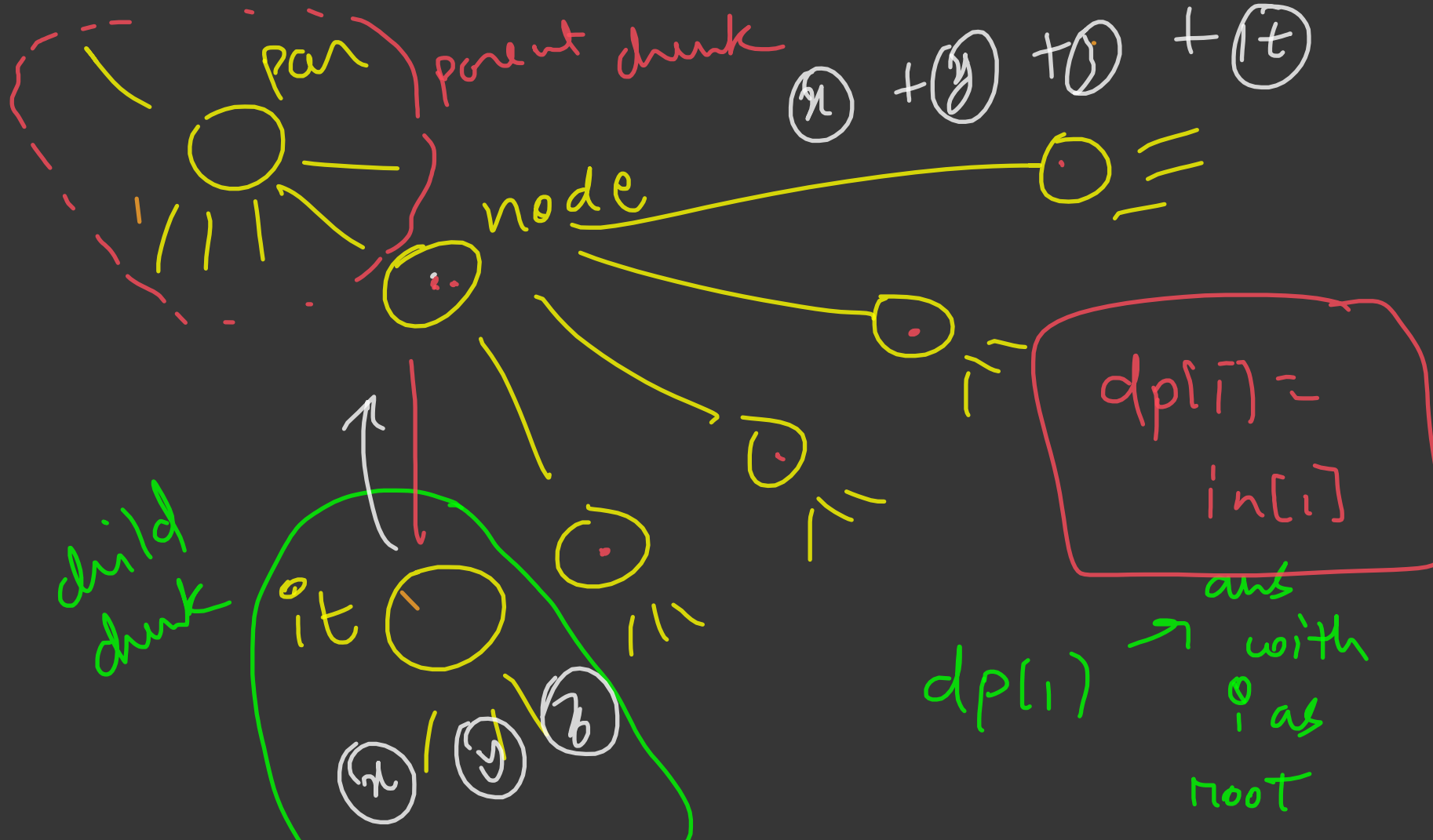
$in[5] = 0$
 $in[4] = 0$
 $in[2] = 0$

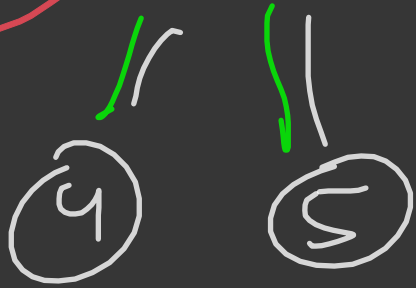
$in[3] = 2$

$$in[1] = in[3] + subtree(3) + in[2] + subtree(2)$$

$in[node] += in[it] + subtree[it];$







subtr

5	1	3	1	1
1	2	3	4	5

$$dp[1] \approx$$
