LCS, LIS, Edit Distance, Staircase, Rock Climbing

Sunday, March 17, 2024 11:03 PM

LCS → Longest Common Sub-Sequence

A → abxy

B → abby

abxy

abby ②

abxy ②

abby

① DP defination ✓

② DP Transition ✓

③ Base Case ✓

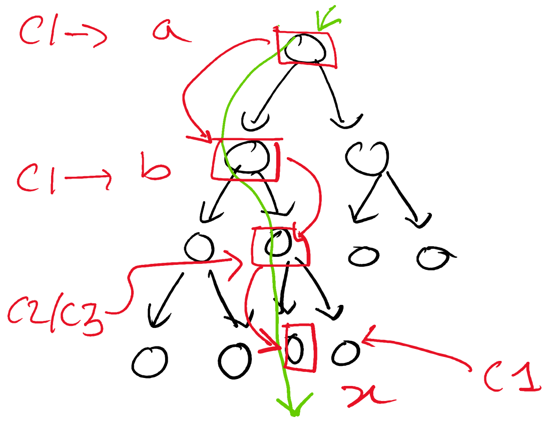
④ Memosization ✓

dp[i][j] = (i → end of A) (j → end of B) LCS

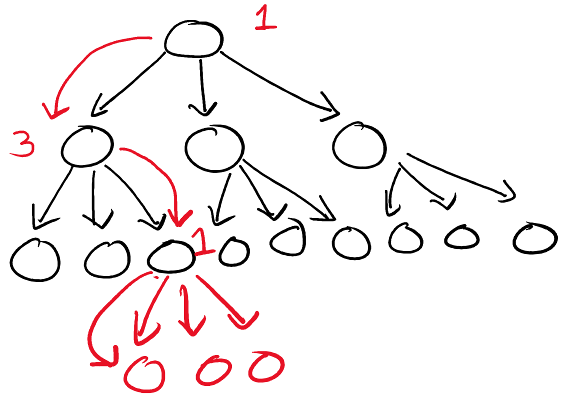
LCS(i, j)

True ↙ ↘ ✗

1+LCS(i+1, j+1) max(LCS(i+1, j), LCS(i, j+1))



abx



LIS → Longest Increasing Sub-sequence

0 1 2 3 4 5 6

$$\begin{array}{ccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 \\ [\,\underline{1}, & 5, & 3, & \underline{2}, & \underline{7}, & 2 & \underline{8}\,] \end{array} \rightarrow 4$$

prevIdx   pos

$$dp\overset{3}{[pos]}\overset{0}{[prevIndex]} \rightarrow [pos, Len) \Rightarrow LIS$$

$$LIS(\overset{1}{pos}, \overset{0}{prevIdx})$$

$$1 + LIS(pos+1, pos) \qquad LIS(pos+1, prevIdx)$$

Rock Climbing

row			
3	4	-5	-2
2	-9	3	4
1	-12	6	-7
0	14	0 ⑧	3
	0	1	2

col

↖ ↑ ↗

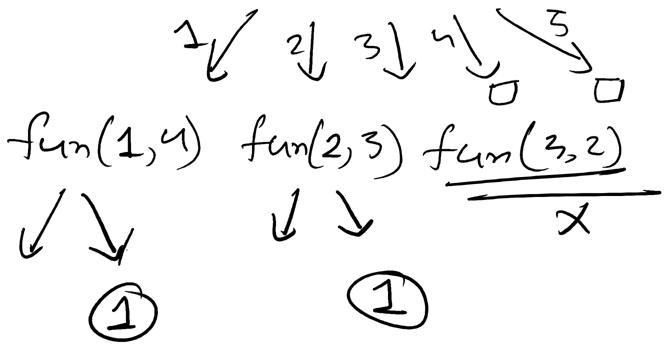
Max Value

(0, x) → Starting Cell

Staircase

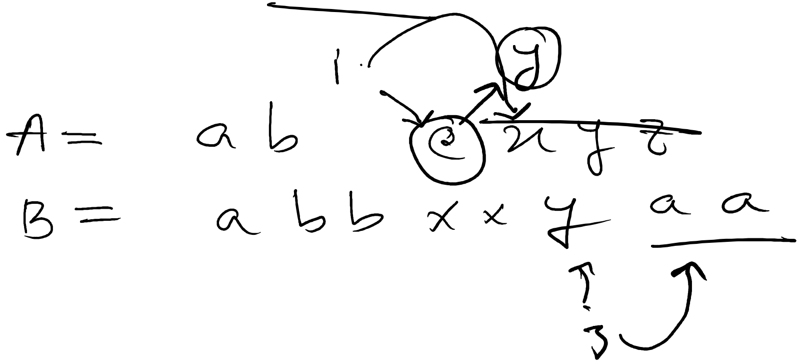
fun(prevStepCnt, rem) → count of Staircases

fun(0, 5)



Edit Distance

Insert, Delete, Replace



String

① Hashing

② KMP

③ Trie / Prefix tree

(Trie + observation + Math)

Software Engineers Role

Coding, debug, bugfix → Junior eng.

API doc