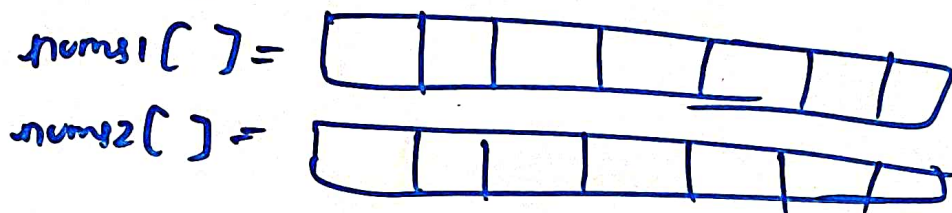


Q) What was my approach and problems did it have?

I tried solving it using dp.



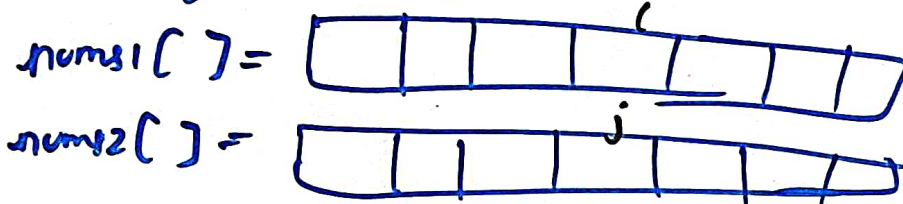
$\{ \text{int}[] \text{nums1}, \text{int idx1}, \text{int}[] \text{nums2}, \text{int idx2}, \text{String cur} \}$
if (idx1 & idx2 have crossed n, m) {
 return cur;
}

if (idx1 < n) {
 \Rightarrow pick nums1[i]
 S1 = f(nums1, idx1+1, nums2, idx2, cur + nums1[i])
 \Rightarrow don't pick nums1[i]
 S2 = f(nums1, idx1+1, nums2, idx2, cur);
 Similarly do it for idx2.

}

Q) What was my approach and problems did it have?

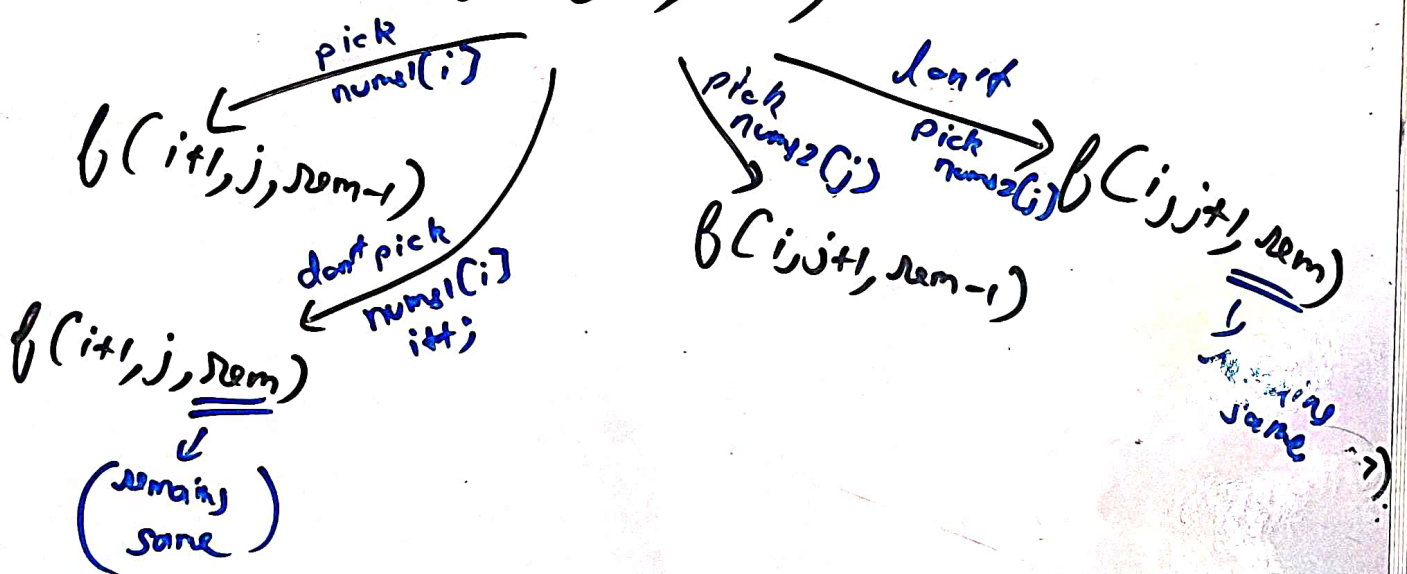
I tried solving it using dp.



rem = k

→ How many characters I can pick.

$f(i, j, \text{rem})$



$$\begin{aligned}
 S_1 &= f(\text{nums}_1, i+1, \text{nums}_2, j, \text{cur} + \text{nums}_1[i], \text{rem}-1) \\
 S_2 &= f(\text{nums}_1, i+1, \text{nums}_2, j, \text{cur}, \text{rem}) \\
 S_3 &= f(\text{nums}_1, i, \text{nums}_2, j+1, \text{cur} + \text{nums}_2[j], \text{rem}) \\
 S_4 &= f(\text{nums}_1, i, \text{nums}_2, j+1, \text{cur}, \text{rem})
 \end{aligned}$$

return $\max(S_1, S_2, S_3, S_4) \Rightarrow$ Since we are constructing the whole string from the beginning so return is the full constructed string.

The problem came when I tried to memoize it or rather cache it.

$dp[i][j][\text{rem}] \rightarrow$ greatest string.

\downarrow

$dp[i+1][j][\text{rem}-1]$
 $dp[i+1][j][\text{rem}]$
 $dp[i][j+1][\text{rem}-1]$
 $dp[i][j+1][\text{rem}]$

} these are the states possible

Problem here is the lack of state string

$\begin{matrix} (i, j, \text{rem}) \text{ can be reached} \\ dp[i][j][\text{rem}][\text{abc}] \\ dp[i][j][\text{rem}][\text{aeb}] \end{matrix} \rightarrow dp[i][j][\text{rem}][\text{str}]$

any way

This becomes a huge space & time complexity.