

18 Feb 2022
Morning 6 AM

Topic : Edit Distance problem

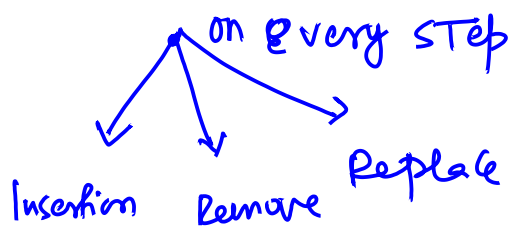
You are given two string `str1` and `str2`. And you can perform 03 possible operation on these strings

- Insert
 - Remove
 - Replace
- } By Using these operation, you need to convert str1 to str2 with minimum number of operation.

@note: All the operation have same cost.

Let's understand problem through Example

Let's understand this.
 Ex ① $\left. \begin{array}{l} \text{str1} = \text{cat} \\ \text{str2} = \text{cut} \end{array} \right\}$ So converting of cat to cut, only need replacement of a to u. Other way also possible but any other way need more number of operation than this.



So, you have 03 option available on every step

So, obvious way of solving this problem using recursion. So you just have to explore all the path until you reach the solution, and save this as a one probable candidate of answer and continue your exploration for any better ^{other} solution possible?

so the approach is recursion with backtrack

Assume that you are reaching to solution by going depth of recursion upto m level than this approach have time complexity $O(m)$.

$T(n): \cancel{O(2^n)} O(3^m)$; m is the recursion depth

str1 str2
Cat cut

if both character are same
Just increase the both pointers

else different
apply Operation Replace/Remove/insert one by one
and call the recursion

So, at the end your task is to find the minimum
operation required.

One tricky part of this problem is to identify
the base case.

I will say when my count of different
character from two string reached 0.

OK, let's discuss the approach

1. Compare the last character from two string, and if
same then ignore the last character and adjust the remaining
string count to like $m-1, n-1$ // ~~where~~ $m \geq n$ is the length
2. if the last character are not same, we consider all
operations on str1, & recursively compute the minimum
cost for all three operations, and take a minimum of three
values.

- Insert : Recur for m and $n-1$
- Remove : Recur for $m-1$ and n
- Replace : Recur for $m-1$ and n

Let's write some pseudocode for our approach.

- if $str1[m] = str2[n]$
 - $editDist(str1, str2, m, n) = editDist(str1, str2, m-1, n-1)$
- else
 - $editDist(str1, str2, m, n) = 1 + \min($
 - $editDist(str1, str2, m-1, n)$ (remove)
 - $editDist(str1, str2, m, n-1)$ (insert)
 - $editDist(str1, str2, m-1, n-1)$

Let's switch on IDE and write your actual code

You will find on the link

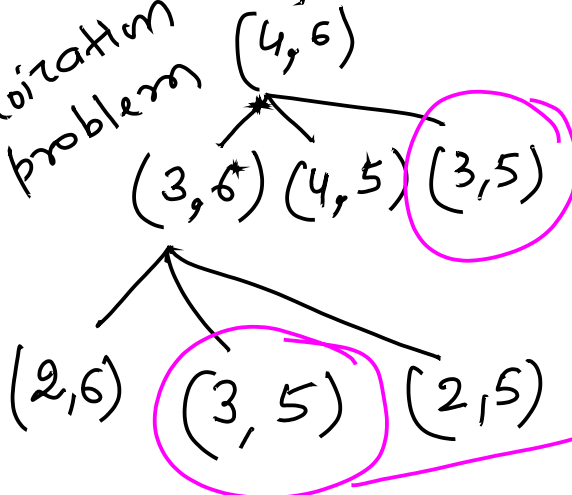
<https://github.com/abonishora1234/4AM-Club-Coding/tree/main/16-Feb-2022>

Just try to find now, does this problem have overlapping substructure?

str1: Sunday (6)
str2: Saturday (8)

(6,8)
(5,7)
(4,6)

So the next is to implement recursion + memoization to solve this problem



overlapping and if actually giving you intuition that dp is better candidate to solve this problem.

for memoization, we are actually need to capture the parameter which is changing across the recursion stack. In this case pointer on one string & pointer on other string and their character comparison is adding operation cost, So you need to handle this for your memoization.

let's write your code in IDE.

✓ Completed and you may find the same in github repository.

Completed the Tabulation based code.

✓ You can try yourself first and then if problem refer my github.

—X—
Thank you.

Pinging off
for this problem.
—X— Completed