st Common subsequence (LCS):

Input: strings $X = x_1 \ldots x_n$ & $Y = y_1 \ldots y_n$

Goal: length of longest string that is a subsequence in both X & Y

Example: X = BCDBCDA
Y = ABECBAA

solution = 4 corresponding to BCBA

Application: UNIX diff

Step 1: Define subproblem.

Attempt 1: Consider prefixes.

For $0 \le i \le n$,

$L(i)$ = length of LCS in $x_1, \ldots, x_i$ with $y_1, \ldots, y_i$

Can we express $L(i)$ in terms of $L(1), \ldots, L(i-1)$?

Earlier example: BCDBCDA
ABECBAA

For $L(7)$: $X_7 = Y_7$ so might as well match

& take best of $L(6)$

So $L(7) = 1 + L(6)$

But what if $X_i \neq Y_i$? E.g. $L(6)$.

Might use X_i or Y_i but not both
then left with diff. length strings.

Attempt 2: For $0 \leq i, j \leq n$,

let $L(i,j)$ = length of LCS of $X_1, \ldots, X_i$
with $Y_1, \ldots, Y_j$

Base cases: $L(i,0)=0$ & $L(0,j)=0$

For recurrence, two cases: $X_i = Y_j$ or $X_i \neq Y_j$

Suppose $X_i = Y_j$:

LCS ends with $X_i = Y_j$ then:

$$L(i,j) = 1 + L(i-1,j-1)$$

otherwise can add $X_i = Y_j$ to make it longer. If X_i matched with Y_ℓ for $\ell < j$ then can match to Y_j for same length.

Hence, $L(i,j) = 1 + L(i-1,j-1)$.

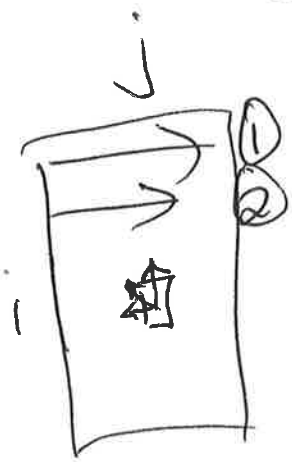
Suppose $X_i \neq Y_j$:

Either X_i not matched &/or Y_j not matched (can't both be matched)

So, $L(i,j) = \max\{L(i-1,j), L(i,j-1)\}$

Summarizing:

$$L(i,j) = \begin{cases} 0 & \text{if } i=0 \text{ or } j=0 \\ 1 + L(i-1,j-1) & \text{if } X_i = Y_j \\ \max\{L(i-1,j), L(i,j-1)\} & \text{if } X_i \neq Y_j \end{cases}$$

$L =$ 

fill table L row-by-row

to get $L(i,j)$:

Use $L(i-1,j), L(i-1,j-1), L(i,j-1)$

LCS(X, Y): $X = x_1, \ldots, x_n$ & $Y = y_1 \cdots y_m$

For $i = 0 \to n$, $L(i,0) = 0$

For $j = 0 \to m$, $L(0,j) = 0$

For $i = 1 \to n$

    For $j = 1 \to m$

        if $X_i = Y_j$

            then $L(i,j) = 1 + L(i-1,j-1)$

            else $L(i,j) = \max\{L(i-1,j), L(i,j-1)\}$

Return$(L(n,m))$

Running time:

For inputs of size n & m,
for loop of size $O(n)$ & inner loop of size $O(m)$
$\Rightarrow O(nm)$ total time.