Longest common subsequence

Subsequence

- A subsequence of a character string $x_0x_1x_2…x_{n-1}$ is a string of the form $x_{i1}x_{i2}… x_{ik}$, where $i_j < i_{j+1}$
- Not the same as substring
- Example
- String: ABCDEFGHIJK
- Subsequence: ACEGJIK
- Subsequence: DFGHK
- Not subsequence: DAGH

The Longest Common Subsequence (LCS) Problem

- Given two strings X and Y, the longest common subsequence (LCS) problem is to find a longest subsequence common to both X and Y
- Has applications to DNA similarity testing (alphabet is {A,C,G,T})
- Example: ABCDEFG and XZACKDFWGH have ACDFG as a longest common subsequence

Brute-force solution

- Enumerate all subsequences of X
- Test which ones are also subsequences of Y
- Pick the longest one.
- Analysis:
- If X is of length n, then it has 2^n subsequences
- This is an exponential-time algorithm!

A Dynamic-Programming Approach to the LCS Problem

- Define L[i,j] to be the length of the longest common subsequence of X[1..i] and Y[1..j].
- Base Case L[0,k] = 0 and L[k,0]=0, to indicate that the null part of X or Y has no match with the other.
- Then define L[i,j] in the general case :

1. If $x_i=y_j$, then L[i,j] = L[i-1,j-1] + 1 (add match)
2. If xi≠yj, then L[i,j] = max{L[i-1,j], L[i,j-1]} (no match)

```
Algorithm LCS(X,Y )
for i = 0 to n do
    L[i,0] = 0
for j = 0 to m do
    L[0,j] = 0
for i =1 to n
    for j =1 to m
    if x_i  = y_j
        L[i, j] = L[i-1, j-1] + 1
    else
        L[i, j] = max{L[i-1, j] , L[i, j-1]}
end
```

Input: Strings X and Y with n and m elements, respectively

Output: For i = 1,…,n, j = 1,...,m, the length L[i, j] of a longest string that is a subsequence of both the string X[0..i] = $x_0x_1x_2…x_i$ and the string Y [0.. j] = $y_0y_1y_2…y_j$

Example : X : optimal Y: similar

L	0	S	I	M	I	L	A	R
0	0	0	0	0	0	0	0	0
O	0	0	0	0	1	1	1	1
P	0	0	0	0	1	1	1	1
T	0	0	0	0	1	1	1	1
I	0	0	0	1	1	1	1	1
M	0	0	1	2	2	2	2	2
A	0	0	1	2	2	2	3	3
l	0	0	1	2	2	3	3	**3**

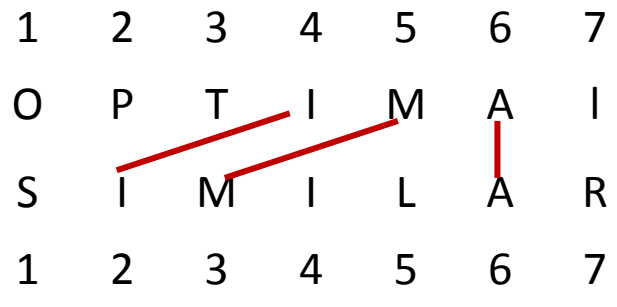
Length of LCS = 3

If $x_i = y_j$ diagonal value +1
Otherwise max(left, up)

LCS – Trace back

L	0	S 1	I 2	M 3	I 4	L 5	A 6	R 7
0	0	0	0	0	0	0	0	0
O 1	0	0	0	0	1	1	1	1
P 2	0	0	0	0	1	1	1	1
T 3	0	0	0	0	1	1	1	1
I 4	0	0	**1**	1	1	1	1	1
M 5	0	0	1	**2**	2	2	2	2
A 6	0	0	1	2	2	2	**3**	3
L 7	0	0	1	2	2	3	3	3

Longest Common Subsequence IMA



Exercise

- Determine the longest Common subsequence of

  X : G T T C C T A A T A

  y : C G A T A A T T G A G A

Analysis of LCS Algorithm

- We have two nested loops
- The outer one iterates n times
- The inner one iterates m times
- A constant amount of work is done inside each iteration of the inner loop
- Thus, the total running time is O(nm)
- Answer is contained in L[n,m] (and the subsequence can be recovered from the L table).