Longest Common Subsequence (LCS)

Subsequence:

Subset of letters in order from left to right.

Remember:

"Subsequence" is different from "substring". Substring is continuous, no gap. But, subsequence can have gap.

Example : Subsequence

X= {A B C B D A B } Y= {B D C A B A}

- CBD is a substring of X. BBA is a subsequence of X, but not a substring of X.
- DBBB is not a subsequence of X, because order violation.
- BDAB is both substring and subsequence of X, but it is only a subsequence of Y

- Substrings are also subsequence, but a subsequence may not be a substring.

Example : LCS

Common Subsequence:

Subsequence of both.

Longest Common Subsequence:

Same subsequence of maximum length.

X= {A B C B D A B }   Y= {B D C A B A}

- X = A BCBD AB,   Y = BD CA BA   LCS: BCBA (4)
- Other examples: BCAB (4),   BDAB (4).
- Maximum possible length = 4

Longest Common Subsequence Problem

Problem:

Given two sequences $X = < x_1, x_2, \ldots\ldots, x_m >$ and
$Y = < y_1, y_2, \ldots\ldots, y_n >$ find a
maximum-length common subsequence of X and Y.

Solution (Idea):

Notice that the LCS problem has *optimal substructure*: solutions of subproblems are parts of the final solution.

Subproblems: "Find LCS of pairs of *prefixes* of X and Y"

Notations

- $X = < x_1, x_2, \ldots\ldots, x_m >$ and $Y = < y_1, y_2, \ldots, y_n >$: Sequences
- X_i, Y_j : Prefixes of X and Y of length i and j respectively
- $c[i,j]$: The length of LCS of X_i and Y_j
- $c[m,n]$: The length of LCS of X and Y

Optimal Substructure of an LCS

- Let $X = < x_1, x_2, \ldots, x_m >$ and $Y = < y_1, y_2, \ldots, y_n >$ be sequences and let $Z = < z_1, z_2, \ldots, z_k >$ be any LCS of X and Y.
- If $x_m = y_n$, then $z_k = x_m = y_n$ and Z_{k-1} is an LCS of X_{m-1} and Y_{n-1}
- If $x_m \neq y_n$, then $z_k \neq x_m$ implies that Z is an LCS of X_{m-1} and Y
- If $x_m \neq y_n$, then $z_k \neq y_n$ implies that Z is an LCS of X and Y_{n-1}

Optimal Substructure of an LCS (Book: Cormen; Page-392)

Let $X = \langle x_1, x_2, \ldots, x_m \rangle$ and $Y = \langle y_1, y_2, \ldots, y_n \rangle$ be sequences, and let $Z = \langle z_1, z_2, \ldots, z_k \rangle$ be any LCS of X and Y.

1. If $x_m = y_n$, then $z_k = x_m = y_n$ and Z_{k-1} is an LCS of X_{m-1} and Y_{n-1}.
2. If $x_m \neq y_n$, then $z_k \neq x_m$ implies that Z is an LCS of X_{m-1} and Y.
3. If $x_m \neq y_n$, then $z_k \neq y_n$ implies that Z is an LCS of X and Y_{n-1}.

Proof (1) If $z_k \neq x_m$, then we could append $x_m = y_n$ to Z to obtain a common subsequence of X and Y of length $k + 1$, contradicting the supposition that Z is a *longest* common subsequence of X and Y. Thus, we must have $z_k = x_m = y_n$. Now, the prefix Z_{k-1} is a length-$(k - 1)$ common subsequence of X_{m-1} and Y_{n-1}. We wish to show that it is an LCS. Suppose for the purpose of contradiction that there exists a common subsequence W of X_{m-1} and Y_{n-1} with length greater than $k - 1$. Then, appending $x_m = y_n$ to W produces a common subsequence of X and Y whose length is greater than k, which is a contradiction.

(2) If $z_k \neq x_m$, then Z is a common subsequence of X_{m-1} and Y. If there were a common subsequence W of X_{m-1} and Y with length greater than k, then W would also be a common subsequence of X_m and Y, contradicting the assumption that Z is an LCS of X and Y.

(3) The proof is symmetric to (2). ∎

The Recurrence

$$c[i, j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ c[i-1, j-1] + 1 & \text{if } i, j > 0 \text{ and } x_i = y_j, \\ \max(c[i, j-1], c[i-1, j]) & \text{if } i, j > 0 \text{ and } x_i \neq y_j \end{cases}$$

LCS Recursive Solution

- We start with $i = j = 0$ (empty substrings of x and y)
- Since X_0 and Y_0 are empty strings, their LCS is always empty (i.e. $c[0,0] = 0$)
- LCS of empty string and any other string is empty, so for every i and j: $c[0, j] = c[i,0] = 0$

LCS Recursive Solution

- When we calculate *c[i,j]*, we consider two cases:

- **First case:** *x[i]=y[j]*: one more symbol matches, so the length of LCS X_i and Y_j equals to the length of LCS of smaller strings X_{i-1} and Y_{i-1} , plus 1
- **Second case:** *x[i] != y[j]:* As symbols do not match, solution is not improved, and the length of LCS(X_i , Y_j) is the same as before (maximum of LCS (X_i, Y_{j-1}) and LCS (X_{i-1},Y_j))

LCS Example

What is the Longest Common Subsequence of X and Y?

X = ABCB

Y = BDCAB

LCS Example (0)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi						
1	**A**						
2	**B**						
3	**C**						
4	**B**						

X = ABCB;   m = |X| = 4
Y = BDCAB; n = |Y| = 5
Allocate array c[6,5]

LCS Example (1)

i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi						
1	**A**	**0**					
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

for i = 1 to m $\quad$ c[i,0] = 0

LCS Example (2)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**					
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

for j = 0 to n $\qquad$ c[0,j] = 0

LCS Example (3)

i		j	0	1	2	3	4	5
		Yj		**B**	**D**	**C**	**A**	**B**
0	Xi		**0**	**0**	**0**	**0**	**0**	**0**
1	**A**		**0**	**0** ↑				
2	**B**		**0**					
3	**C**		**0**					
4	**B**		**0**					

case i=1 and j=1

A != B

but, c[0,1]>=c[1,0]

so c[1,1] = c[0,1], and b[1,1] = ↑

LCS Example (4)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑			
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

case i=1 and j=2

A != D

but, c[0,2]>=c[1,1]

so c[1,2] = c[0,2], and b[1,2] = ↑

LCS Example (5)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑		
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

case i=1 and j=3

A != C

but, c[0,3]>=c[1,2]

so c[1,3] = c[0,3], and b[1,3] = ↑

LCS Example (6)

	j	0	1	2	3	4	5
i		Yj	B	D	C	A	B
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	A	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	
2	B	**0**					
3	C	**0**					
4	B	**0**					

case i=1 and j=4

A = A

so c[1,4] = c[0,2]+1, and b[1,4] = ↖

LCS Example (7)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

case i=1 and j=5

A != B

this time c[0,5]<c[1,4]

so c[1,5] = c[1, 4], and b[1,5] = ←

LCS Example (8)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖				
3	**C**	**0**					
4	**B**	**0**					

case i=2 and j=1

B = B

so c[2, 1] = c[1, 0]+1, and b[2, 1] = ↖

LCS Example (9)

		j	0	1	2	3	4	5
i			Yj	B	D	C	A	B
0	Xi		0	0	0	0	0	0
1	A		0	0 ↑	0 ↑	0 ↑	1 ↖	1 ←
2	B		0	1 ↖	1 ←			
3	C		0					
4	B		0					

case i=2 and j=2

B != D

and c[1, 2] < c[2, 1]

so c[2, 2] = c[2, 1] and b[2, 2] = ←

LCS Example (10)

i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←		
3	**C**	**0**					
4	**B**	**0**					

case i=2 and j=3

B != D

and c[1, 3] < c[2, 2]

so c[2, 3] = c[2, 2] and b[2, 3] = ←

LCS Example (11)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	
3	**C**	**0**					
4	**B**	**0**					

case i=2 and j=4

B != A

and c[1, 4] = c[2, 3]

so c[2, 4] = c[1, 4] and b[2, 2] = ↑

LCS Example (12)

i	j	0	1	2	3	4	5
	Yj	B	D	C	A	B	
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**	**0**					
4	**B**	**0**					

case i=2 and j=5

B = B

so c[2, 5] = c[1, 4]+1 and b[2, 5] = ↖

LCS Example (13)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**	**0**	**1** ↑				
4	**B**	**0**					

case i=3 and j=1

C != B

and c[2, 1] > c[3,0]

so c[3, 1] = c[2, 1] and b[3, 1] = ↑

LCS Example (14)

	j	0	1	2	3	4	5
i		Yj	B	D	C	A	B
0	Xi	0	0	0	0	0	0
1	A	0	0 ↑	0 ↑	0 ↑	1 ↖	1 ←
2	B	0	1 ↖	1 ←	1 ←	1 ↑	2 ↖
3	C	0	1 ↑	1 ↑			
4	B	0					

case i=3 and j= 2

C != D

and c[2, 2] = c[3, 1]

so c[3, 2] = c[2, 2] and b[3, 2] = ↑

LCS Example (15)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**	**0**	**1** ↑	**1** ↑	**2** ↖		
4	**B**	**0**					

case i=3 and j= 3

C = C

so c[3, 3] = c[2, 2]+1 and b[3, 3] = ↖

LCS Example (16)

		j	0	1	2	3	4	5
i			Yj	**B**	**D**	**C**	**A**	**B**
0	Xi		**0**	**0**	**0**	**0**	**0**	**0**
1	**A**		**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**		**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**		**0**	**1** ↑	**1** ↑	**2** ↖	**2** ←	
4	**B**		**0**					

case i=3 and j= 4

C != A

c[2, 4] < c[3, 3]

so c[3, 4] = c[3, 3] and b[3, 3] =

LCS Example (17)

i	Xi	j = 0 Yj	1 B	2 D	3 C	4 A	5 B
0		0	0	0	0	0	0
1	A	0	0 ↑	0 ↑	0 ↑	1 ↖	1 ←
2	B	0	1 ↖	1 ←	1 ←	1 ↑	2 ↖
3	C	0	1 ↑	1 ↑	2 ↖	2 ←	2 ↑
4	B	0					

case i=3 and j= 5

C != B

c[2, 5] = c[3, 4]

so c[3, 5] = c[2, 5] and b[3, 5] = ↑

LCS Example (18)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**	**0**	**1** ↑	**1** ↑	**2** ↖	**2** ←	**2** ↑
4	**B**	**0**	**1** ↖				

case i=4 and j=1

B = B

so c[4, 1] = c[3, 0]+1 and b[4, 1] = ↖

LCS Example (19)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**	**0**	**1** ↑	**1** ↑	**2** ↖	**2** ←	**2** ↑
4	**B**	**0**	**1** ↖	**1** ↑			

case i=4 and j=2

B != D

c[3, 2] = c[4, 1]

so c[4, 2] = c[3, 2] and b[4, 2] = ↑

LCS Example (20)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**	**0**	**1** ↑	**1** ↑	**2** ↖	**2** ←	**2** ↑
4	**B**	**0**	**1** ↖	**1** ↑	**2** ↑		

case i=4 and j= 3

B != C

c[3, 3] > c[4, 2]

so c[4, 3] = c[3, 3] and b[4, 3] = ↑

LCS Example (21)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**	**0**	**1** ↑	**1** ↑	**2** ↖	**2** ←	**2** ↑
4	**B**	**0**	**1** ↖	**1** ↑	**2** ↑	**2** ↑	

case i=4 and j=4

B != A

c[3, 4] = c[4, 3]

so c[4, 4] = c[3, 4] and b[3, 5] = ↑

LCS Example (22)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0** ↑	**0** ↑	**0** ↑	**1** ↖	**1** ←
2	**B**	**0**	**1** ↖	**1** ←	**1** ←	**1** ↑	**2** ↖
3	**C**	**0**	**1** ↑	**1** ↑	**2** ↖	**2** ←	**2** ↑
4	**B**	**0**	**1** ↖	**1** ↑	**2** ↑	**2** ↑	**3** ↖

case i=4 and j=5

B= B

so c[4, 5] = c[3, 4]+1 and b[4, 5] = ↖

Algorithm: LCS

```
LCS-Length(X, Y)
m = length(X),  n  = length(Y)
for i = 1 to m
      c[i, 0] = 0
for j = 0 to n
      c[0, j] = 0
for i = 1 to m
         for j = 1 to n
               if ( x_i = = y_j )
                      c[i, j] = c[i - 1, j - 1] + 1
                      b[i, j] = "↖"
               else if c[i - 1, j]>=c[i,  j - 1]
                      c[i, j] = c[i - 1, j]
                      b[i, j] = "↑"
               else c[i, j] = c[i, j - 1]
                      b[i, j] = "←"
return c and b
```

Finding LCS

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

Finding LCS (2)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

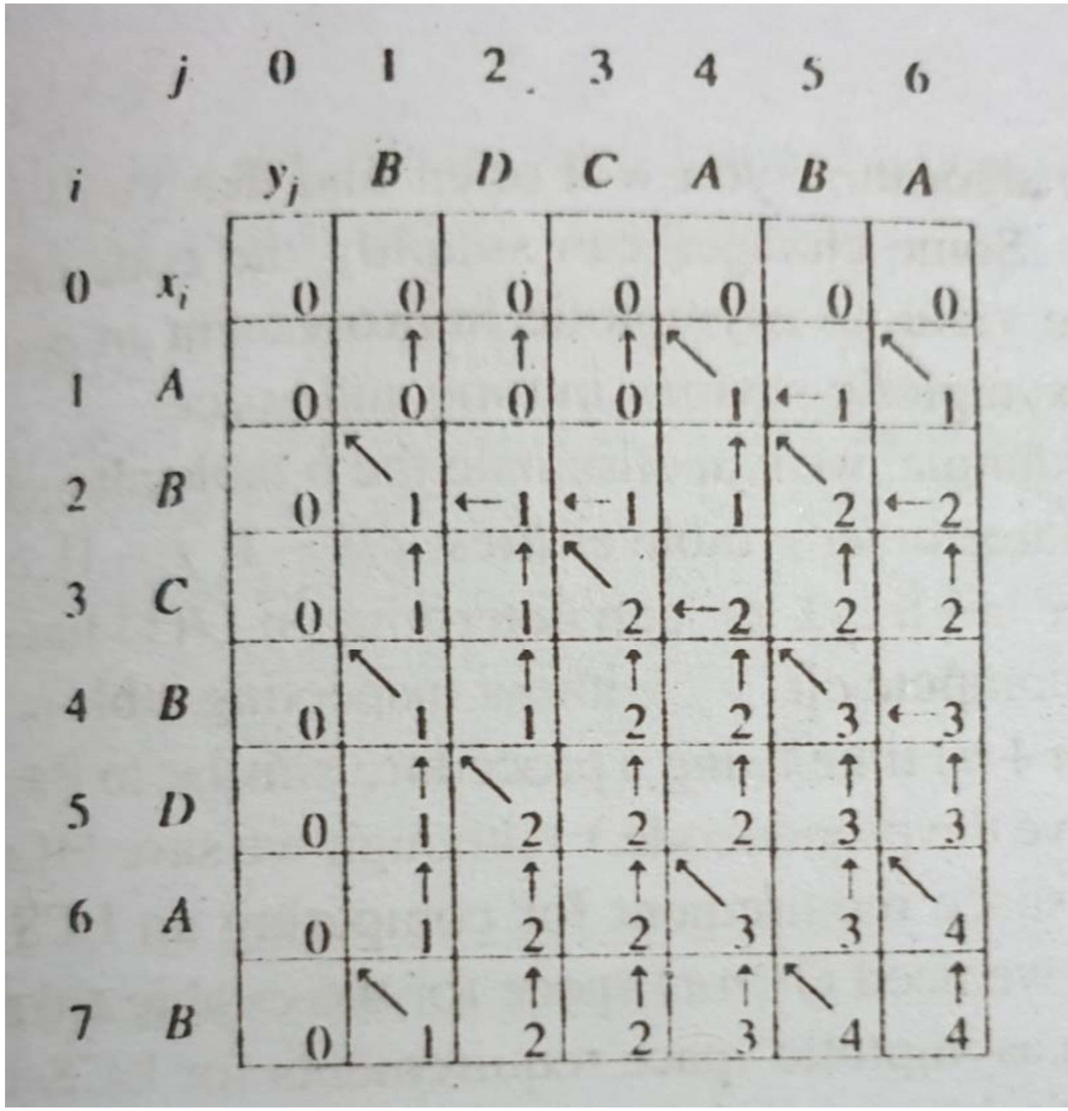
LCS (reversed order): **B C B**

LCS (straight order): **B C B**

Algorithm: Finding LCS

```
PRINT-LCS(b, X, i, j)
if i == 0 or j == 0
    return
if b[i, j] == "↖"
    PRINT-LCS(b, X, i − 1, j − 1)
    print x_i
elseif b[i, j] == "↑"
    PRINT-LCS(b, X, i − 1, j)
else PRINT-LCS(b, X, i, j − 1)
```

Practice Example (Book: Cormen; Page-395)



i	j	0	1	2	3	4	5	6
	y_j		B	D	C	A	B	A
0	x_i	0	0	0	0	0	0	0
1	A	0	↑ 0	↑ 0	↑ 0	↖ 1	← 1	↖ 1
2	B	0	↖ 1	← 1	← 1	↑ 1	↖ 2	← 2
3	C	0	↑ 1	↑ 1	↖ 2	← 2	↑ 2	↑ 2
4	B	0	↖ 1	↑ 1	↑ 2	↑ 2	↖ 3	← 3
5	D	0	↑ 1	↖ 2	↑ 2	↑ 2	↑ 3	↑ 3
6	A	0	↑ 1	↑ 2	↑ 2	↖ 3	↑ 3	↖ 4
7	B	0	↖ 1	↑ 2	↑ 2	↑ 3	↖ 4	↑ 4

LCS : BCBA

Length : 4

Thank You

Stay Safe