



Μέγιστη κοινή υπακολουθία (LCS: Longest common subsequence)

- Ακολουθία $X = \langle x_1, x_2, \dots, x_m \rangle$
- Ακολουθία $Y = \langle y_1, y_2, \dots, y_n \rangle$

- Υπακολουθία $Z = \langle z_1, z_2, \dots, z_k \rangle$

- ***k* μέγιστο**

■ Instance: $X = \langle A, B, C, B, D, A, B \rangle$
 $Y = \langle B, D, C, A, B, A \rangle$

$\langle B, C, A \rangle$



Εφαρμογές: σύγκριση οργανισμών

- Βάσεις τμημάτων DNA

- Adenine – αδενίνη (A)
- Cytosine – κυτοσίνη (C)
- Guanine – γουανίνη (G)
- Thymine – θυμίνη (T)

{A, C, G, T}

- S1 = ACCGGTCGAGTGCGCGGAAGCCGGCCGAA

- S2 = GTCGTTCGGAATGCCGTTGCTCTGTAAA



Εφαρμογές: σύγκριση οργανισμών

- Ομοιότητα

- S1 = ACCGGTCGAGTGCGCGGAAGCCGGCCGAA

- S2 = GTCGTTCTGGAATGCCGTTGCTCTGTAAA

- S1 υποσύνολο S2

- S1 → S2 or S2 → S1

- S3 = GTCGTCGGAAGCCGGCCGAA (μέγιστη κοινή υπακολουθία)



Dynamic Programming

- Ορισμός υπο-προβλημάτων
- Σύνδεση βέλτιστων λύσεων (αναδρομική σχέση)
- Κατασκευή του πίνακα βέλτιστων λύσεων



Ορισμός υποπροβλημάτων

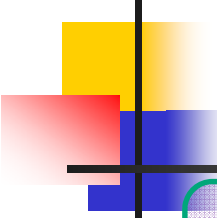
- Αν $x_m = y_n$, $z[k] = x_m = y_n$ πρέπει να βρούμε μια *CLS* $z[k-1]$ των X_{m-1} και Y_{n-1}
- Αν $x_m \neq y_n$, $Z[k] \neq x_m$ πρέπει να βρούμε μια *CLS* των X_{m-1} και y_n
- Αν $x_m \neq y_n$, $Z[k] \neq y_n$ πρέπει να βρούμε μια *CLS* των x_m και Y_{n-1}

Δυναμικός Προγραμματισμός (παράδειγμα) / υποπροβλήματα

	0	1	2	.	.	j	.	.	.	n-1	n
0											
1											
.											
i											
.											
m-1											
m											

$f(m,n)$: τιμή βέλτιστης λύσης

$f(m-1, n-1), f(m, n-1), f(m-1, n)$



Δυναμικός Προγραμματισμός/LCS Αναδρομική σχέση

- $f[i, j]$ το μήκος μιας LCS των ακολουθιών $X[i]$ και $Y[j]$

- If $i = 0$ or $j = 0$, τότε η LCS είναι 0 και

- $f[i, j] =$
 - 0 if $i=0$ or $j = 0$
 - $f[i - 1, j - 1] + 1$, if $i, j > 0$ and $x[i] = y[j]$
 - $\max (f[i, j - 1], f[i - 1, j])$ if $i, j > 0$ and $x[i] \neq y[j]$



Αναδρομικός Αλγόριθμος (LCS)

- LCS (x,y,i,j)

```
if x[i]=y[j] then f[i,j]= LCS(x,y,i-1,j-1) +1
```

```
else
```

```
f[i,j]=max{LCS(x,y,i-1,j),  
LCS(x,y,i,j-1)}
```

```
return f
```


Δυναμικός Προγραμματισμός/LCS

εύρεση τιμής

LCS (X, Y)

$m = \text{length}[X], n = \text{length}[Y]$

for $i = 1$ to m do $f[i, 0] = 0$

for $j = 0$ to n do $f[0, j] = 0$

for $i = 1$ to m

for $j = 1$ to n

if $x[i] = y[j]$ then $f[i, j] = f[i - 1, j - 1] + 1$

else if $f[i - 1, j] \geq f[i, j - 1]$

then $f[i, j] = f[i - 1, j]$

else $f[i, j] = f[i, j - 1]$

return f

Δυναμικός Προγραμματισμός/LCS εύρεση και της δομής της λύσης

LCS (X, Y)

```
 $m = \text{length}[X], n = \text{length}[Y]$   
for  $i = 1$  to  $m$  do  $f[i, 0] = 0$   
for  $j = 0$  to  $n$  do  $f[0, j] = 0$ 
```

```
for  $i = 1$  to  $m$   
  for  $j = 1$  to  $n$ 
```

```
    if  $x[i] = y[j]$  then  $f[i, j] = f[i - 1, j - 1] + 1$   
       $\text{path}[i, j] = 'D'$ 
```

```
    else if  $f[i - 1, j] \geq f[i, j - 1]$   
      then  $f[i, j] = f[i - 1, j], \text{path}[i, j] = 'U'$   
      else  $f[i, j] = f[i, j - 1], \text{path}[i, j] = 'L'$ 
```

```
return  $f, \text{path}$ 
```

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A
 X[i]: A B C B D A B

	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						
2 B	0						
3 C	0						
4 B	0						
5 D	0						
6 A	0						
7 B	0						

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A
 X[i]: A B C B D A B

	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						
3 C	0						
4 B	0						
5 D	0						
6 A	0						
7 B	0						

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A
 X[i]: A B C B D A B

	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
		U	U	U	D	L	D
1 A	0	0	0	0	1	1	1
		D	L	L	U	D	L
2 B	0	1	1	1	1	2	2
3 C	0						
4 B	0						
5 D	0						
6 A	0						
7 B	0						

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A

X[i]: A B C B D A B

	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
		U	U	U	D	L	D
1 A	0	0	0	0	1	1	1
		D	L	L	U	D	L
2 B	0	1	1	1	1	2	2
		U	U	D	L	U	U
3 C	0	1	1	2	2	2	2
4 B	0						
5 D	0						
6 A	0						
7 B	0						

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A
 X[i]: A B C B D A B

	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
		U	U	U	D	L	D
1 A	0	0	0	0	1	1	1
		D	L	L	U	D	L
2 B	0	1	1	1	1	2	2
		U	U	D	L	U	U
3 C	0	1	1	2	2	2	2
		D	U	U	U	D	L
4 B	0	1	1	2	2	3	3
5 D	0						
6 A	0						

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A
 X[i]: A B C B D A B

	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
		U	U	U	D	L	D
1 A	0	0	0	0	1	1	1
		D	L	L	U	D	L
2 B	0	1	1	1	1	2	2
		U	U	D	L	U	U
3 C	0	1	1	2	2	2	2
		D	U	U	U	D	L
4 B	0	1	1	2	2	3	3
		U	D	U	U	U	U
5 D	0	1	2	2	2	3	3

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A
 X[i]: A B C B D A B

	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	U/0	U/0	U/0	D/1	L/1	D/1
2 B	0	D/1	L/1	L/1	U/1	D/2	L/2
3 C	0	U/1	U/1	D/2	L/2	U/2	U/2
4 B	0	D/1	U/1	U/2	U/2	D/3	L/3
5 D	0	U/1	D/2	U/2	U/2	U/3	U/3
6 A	0	U/1	U/2	U/2	D/3	U/3	D/4
7 B	0						

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A
 X[i]: A B C B D A B

	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	U/0	U/0	U/0	D/1	L/1	D/1
2 B	0	D/1	L/1	L/1	U/1	D/2	L/2
3 C	0	U/1	U/1	D/2	L/2	U/2	U/2
4 B	0	D/1	U/1	U/2	U/2	D/3	L/3
5 D	0	U/1	D/2	U/2	U/2	U/3	U/3
6 A	0	U/1	U/2	U/2	D/3	U/3	D/4
7 B	0	D/1	U/2	U/2	U/3	D/4	U/4

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση της τιμής

Y[j]: B D C A B A
 X[i]: A B C B D A B

	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	U/0	U/0	U/0	D/1	L/1	D/1
2 B	0	D/1	L/1	L/1	U/1	D/2	L/2
3 C	0	U/1	U/1	D/2	L/2	U/2	U/2
4 B	0	D/1	U/1	U/2	U/2	D/3	L/3
5 D	0	U/1	D/2	U/2	U/2	U/3	U/3
6 A	0	U/1	U/2	U/2	D/3	U/3	D/4
7 B	0	D/1	U/2	U/2	U/3	D/4	U/4

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / εύρεση δομής της λύσης

Y[j]: B D C A B A
 X[i]: A B C B D A B

	0	1	2	3	4	5	6
	0	B	D	C	A	B	A
0 x[i]	0	0	0	0	0	0	0
1 A	0	U/0	U/0	U/0	D/1	L/1	D/1
2 B	0	D/1	L/1	L/1	U/1	D/2	L/2
3 C	0	U/1	U/1	D/2	L/2	U/2	U/2
4 B	0	D/1	U/1	U/2	U/2	D/3	L/3
5 D	0	U/1	D/2	U/2	U/2	U/3	U/3
6 A	0	U/1	U/2	U/2	D/3	U/3	D/4
7 B	0	D/1	U/2	U/2	U/3	D/4	U/4

LCS: Δυναμικός Προγραμματισμός (παράδειγμα) / Πολυπλοκότητα

Χρόνος

- $O(mn)$ για εύρεση της τιμής
- $O(m+n)$ για εύρεση της δομής της λύσης

Μνήμη

- $O(mn)$ για εύρεση της τιμής
- $O(mn)$ για εύρεση της δομής της λύσης