Longest Common Sequence

String1 a b c d e f g h i j

String2 c d g i

→ In Kmp or other, we have substrings.

→ We are not looking for exact match

→ we need which of the above are from seq1

a b c d e f g h i j

c d g i

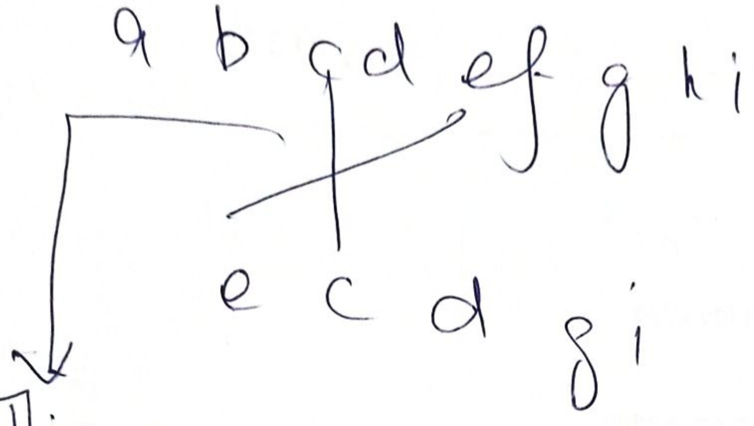
cdgi → exists even not together different.

↳ longest common subsequence

dgi

gi

Change in example



This cross is not allowed

no intersection

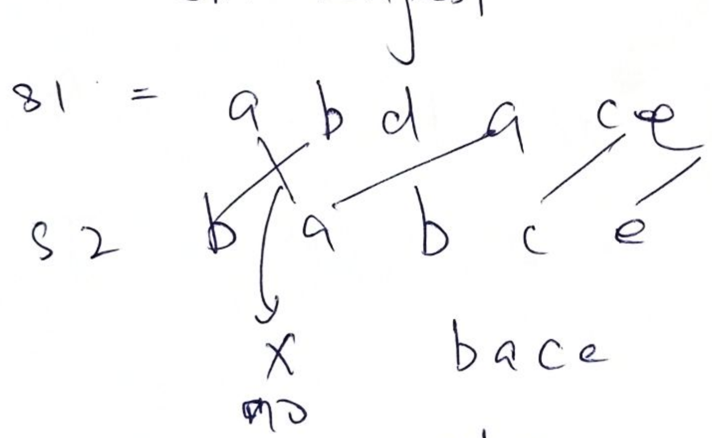
c should follow e

egi → 1 subsequence.

cdgi → another subsequence

↓

still longest.



bace 1 sequence

abce another

So there can be multiple

Subsequence, matching may

not be continous

If we do it with recursion

```
if LCS(i j)
{
  if A[i] == '\0' || B[j] == '\0'
      return zero
  else if (A[i] == A[j])
      return 1 + LCS[i+1, j+1]
  else
      return max(LCS[i+1, j]
                 LCS(i, j+1)
```

A - b d \0

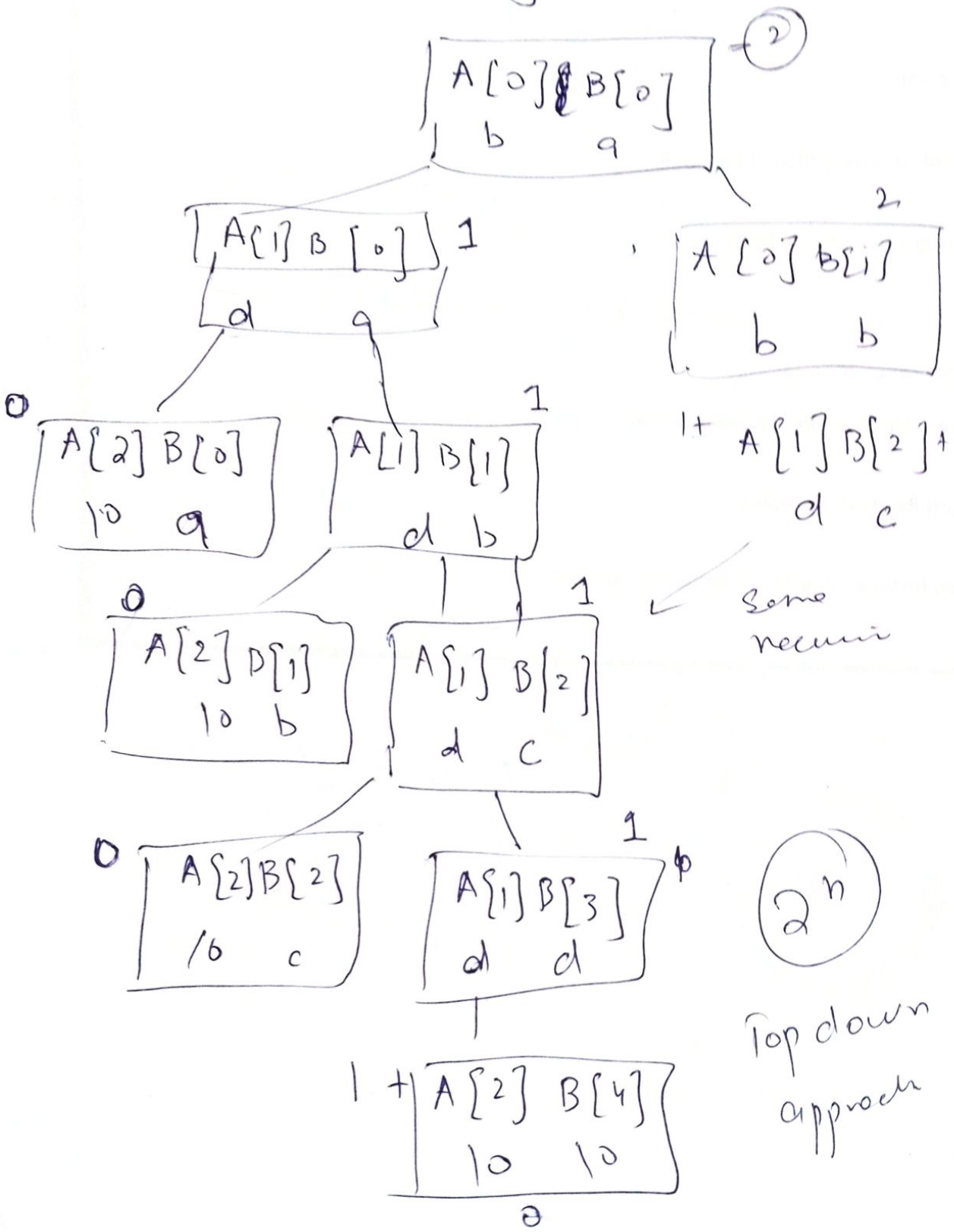
B a b c d \0

B = b d \0

B = a b c d \0

(4)

Tracing



Now dp

if (A[i] = B[j]) (5)

LCS[i, j] = 1 + LCS[[i-1], j-1]

else

LCS[i, j] = max(LCS[[i-1], j], LCS[i, j-1])

A | b | d |

B | a | b | c | d |

			a	b	c	d
		0	1	2	3	4
	0	0	0	0	0	0
b	1	0	0	1	1	1
d	2	0	0	1	1	2

so d

so b

mxn

Str2 stone

Str1 longest

			l	o	n	g	e	s	t
		0	1	2	3	4	5	6	7
	0	0	0	0	0	0	0	0	0
s	1	0	0	0	0	0	0	1	1
t	2	0	0	0	0	0	0	1	2
o	3	0	0	1	1	1	1	1	2
n	4	0	0	1	2	2	2	2	2
e	5	0	0	1	2	2	3	3	3

o n e

m×n

S1 ⇒ b a b b a b

S2 a b a a b a

	b	a	b	b	a	b
a	0	1	1	1	1	1
b	1	1	2	2	2	2
a	1	2	2	2	3	3
a	1	2	3	2	3	3
b	1	2	3	3	3	4
a	1	2	3	3	4	4

baba