Sub optimality Property for LCS

Common SubSequence of 2 Sequences:

$$A_m = \langle a_1, a_2, \ldots, a_m \rangle$$
$$B_n = \langle b_1, b_2, \ldots, b_n \rangle$$

if $A[m] \neq B[n]$: at most one belongs to a common subsequence

1) if $A[m]$ is not in LCS
$A_{m-1} = \langle a_1, a_2, \ldots a_{m-1} \rangle$
$B_n = \langle b_1, b_2, \ldots, b_n \rangle$

2) if $B[n]$ is not in LCS
$A_m = \langle a_1, a_2, \ldots, a_m \rangle$
$B_{n-1} = \langle b_1, b_2, \ldots, b_{n-1} \rangle$

Given OPT = LCS
Subset Problem: S_a give sequence in $A_m \in S$ let $S_a \subset S$ by the set of longest common sequence

Table C: $(m+1) \times (n+1)$

SO_i $(O_i \in OPT)$

3 problems

1) if $A[i] == B[j]$
$C[i, j] = C[i-1, j-1] + 1$
$b[i,j] \nwarrow$

2) $C[i,j]$ $C[i-1, j]$ $\uparrow$

$C[i-1,j] \geq C[i, j-1]$

3) $\leftarrow$

1) Suppose common sequence X
Then $X[A_{m-1}, B_{n-1}] >$ ~~OPT LCS length~~
$X[A_m, B_n] = X'$

$|X| + \{O_i\} > |X'| + \{O_i\}$

Proof by contradiction,
$X[A_m, B_n]$ would be larger than OPT LCS length

2) $X[A_{m-1}, B_n]$ is OPT then $X[A_m, B_n]$ is common sequence $C[i,j]$ but $C[i,j] = C[i-1, j]$

3) $\{O_i\}$ $X[A_m, B_{n-1}]$ is OPT
but $X[A_m, B_n] > X[A_m, B_{n-1}] >$ OPT $\{O_i\}$

2 max Subarray $A[0 \ldots n-1]$ i ≤ j Subarray $A[i \ldots j]$
has greatest Sum of any nonempty subarray of A

1) a) Greedy choice: update maxSum if CurrSum is larger
b) maxSum=0; CurrSum=0; update currSum if $A_i >$ currSum

```
for i in len(A)
    currSum= max(currSum+i, i)
    maxSum= max(maxSum, CurrSum)
return maxSum
```

c) O(n)

2) a) if Prev element >0 update curr with prev Sum
b)

```
for i len(A)
    if A[i]>0
        curr = curr+ A[i-1]
    maxSum= (maxSum, num)
return maxSum
```

c) O(n)

3) input seq n words of lengths $l_1, l_2, l_3 \ldots$
# of lines that hold a max of M char
if $i \leq j$ & 1 space end of the line is $M - j + i - \sum_{k=i}^{j} l_k$
minimize sum. Sum = Cost of Printing

n = sequence of words $\quad$ m = max characters per line
extra space @ EOL = $m - j + i \sum_{k=i}^{j} l_k$
Words on line = (i, j)
Set $S(i, j)$ = words $\quad$ maxSum = Cost of Printing arr?

Recursive $C[i] = \max((n-1) + lc[i, j])$
$\max\ m[i+1]$
max $C[0] = 0$
$C[j] = \min(S(i) + C[i, j])$

$O(n^2)$