O PROBLEMA (definição de outra fonte)

Seja que $A[1 \dots n]$ é uma sequência de números. Uma subsequência de $A[1 \dots n]$ é o que sobra depois que um conjunto arbitrário de elementos da sequência é apagado.

Definição mais formal: Uma sequência $S[1 \dots k]$ é subsequência de uma sequência $A[1 \dots k]$ se existe uma injeção de S em A. Uma injeção de S em A é uma sequência $(i_1, i_2, \dots i_k)$ de índices tal que $1 \le i_1 < i_2 < \dots < i_k \le n$ e

$$S[1] = A[i_1],\ S[2] = A[i_2], \dots,\ S[k] = A[i_k]$$

Dizemos que um índice j de S casa com um índice m de A se existe uma injeção $(i_1, i_2, \dots, i_k)$ tal que $i_j = m$.

SUBSEQUÊNCIAS (definição do slide)

$\langle z_1, \dots, z_k \rangle$ é subsequência de $\langle x_1, \dots, x_m \rangle$ se existem índices $i_1 < \dots < i_k$ tais que

$$z_1 = x_{i_1} \quad \dots \quad z_k = x_{i_k}$$

ANOTAÇÕES (da aula)

- problema: encontrar a subsequência comum mais longa entre A e B
- diff: as linhas sem sinais mostram a <u>ssco</u>
- EXEMPLO:

X = ABRACADABRA

Y = ABACATE

Z = A ...

- se X e Y começam com a mesma letra, Z também começará, o mesmo vale quando terminam com a mesma letra.

$X[1 \dots m]$ $Y[1 \dots n]$

se $X[m] = Y[n]$

então $Z[k] = X[m]$

$Z[1 \dots k-1]$ é LCS de $X[1 \dots m-1]$ e $Y[1 \dots n-1]$

se $X[m] \neq Y[n]$

então $Z[1 \ldots k]$ é LCS

ou de $X[1 \ldots m-1]$ e $Y[1 \ldots n]$

ou de $X[1 \ldots m]$ e $Y[1 \ldots n-1]$

- problema: encontrar o comprimento de uma ssco máxima.
- consumo de tempo:

$X = A^{20}$

$Y = A^{20}$

$c[20,20]$

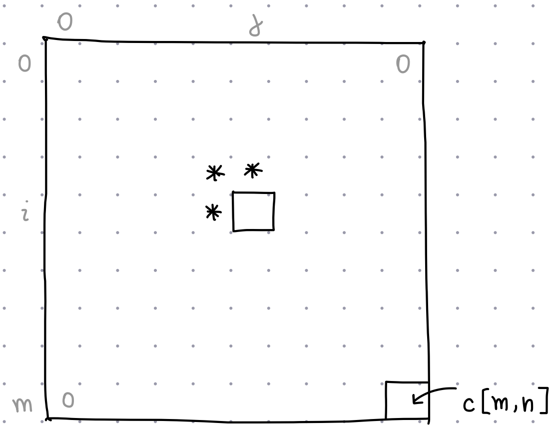
$c[19,20]$ $c[20,19]$

$c[18,20]$ $c[19,19]$ $c[19,19]$ $c[20,18]$

tempo da chamada com $n = m$

$T(n) \geq 2T(n-1) + 1 \implies T(n) = \Omega(2^n)$

- o algoritmo faz $n \times m$ chamadas com i e j diferentes em $c[i,j]$
- truncamento:
  - tamanho da tabela: $(n+1)(m+1)$
  - custo para preencher uma entrada da tabela considerando que as chamadas recursivas custam $O(1)$
  - $(n+1)(m+1) \cdot O(1) = \Theta(nm)$
- ordem para preencher a matriz
- percorrer por linha ou coluna

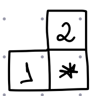


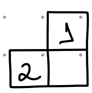
preenchimento da matriz

- diferentes: máximo entre o de cima e o do lado
- iguais: pega a diagonal e soma 1

encontrar a LCS: rastrear os casamentos → casamento: (i,j) da matriz que as letras são iguais

- quadrático para preencher a matriz
- linear para percorrer e achar a LCS.

- se é igual: ↖
- se o de cima é maior que o da esquerda  : ↑

- se é menor  : ←