

Pregunta 2.

a) Selection Sort (A, n)

for $i = 1, \dots, n-1$

smallest = i

for $j = i+1 \dots n$

if $A[j] < A[\text{smallest}]$

smallest = j

intercambiar $A[i]$ con $A[\text{smallest}]$

Como cada elemento de A es otro arreglo, tenemos que comparar el segundo término del arreglo.

Selection Sort (A, n)

for $i = 1, \dots, n-1$

smallest = i

for $j = i+1 \dots n$

if $A[j][1] < A[\text{smallest}][1]$

smallest = j

intercambiar $A[i]$ con $A[\text{smallest}]$

tenemos que por cada elemento (partiendo por (yo, 23)) designamos el smallest como el índice, al ir comparando, llega a la primera iteración tendremos.

$[(\text{él}, 2), (\text{tú}, 20), (\text{yo}, 23), \dots]$ # todo lo demás igual.

segunda iteración (tomamos $i = (\text{tú}, 20)$).

$[(\text{él}, 2), (\text{vosotras}, 5), (\text{yo}, 23), (\text{ella}, 10), (\text{nosotros}, 25), (\text{vosotros}, 15), (\text{ellos}, 25), (\text{ellas}, 30),$

$(\text{ustedes}, 10), (\text{tú}, 20), (\text{todos}, 12)]$

tercera iteración ($i = (\text{yo}, 23)$).

$[(\text{él}, 2), (\text{vosotras}, 5), (\text{ella}, 10), (\text{yo}, 23), (\text{nosotros}, 25) \dots]$ # todo lo demás igual

cuarta iteración ($i = (\text{yo}, 23)$)

$[(\text{él}, 2), (\text{vosotras}, 5), (\text{ella}, 10), (\text{ustedes}, 10), (\text{nosotros}, 25), (\text{vosotros}, 15), (\text{ellos}, 25), (\text{ellas}, 30),$

$(\text{yo}, 23), (\text{tú}, 20), (\text{todos}, 12)]$

5) $[(\text{él}, 2), (\text{vosotras}, 5), (\text{ella}, 10), (\text{ustedes}, 10), (\text{todos}, 12), (\text{vosotros}, 15), (\text{ellos}, 25), (\text{ellas}, 30),$

$(\text{yo}, 23), (\text{tú}, 20), (\text{nosotros}, 25)]$

6) no cambia

7) $[(\text{él}, 2), (\text{vosotras}, 5), (\text{ella}, 10), (\text{ustedes}, 10), (\text{todos}, 12), (\text{vosotros}, 15), (\text{tú}, 20), (\text{yo}, 23), (\text{ellas}, 30),$

$(\text{yo}, 23), (\text{ellos}, 25), (\text{nosotros}, 25)]$

8) $[(\text{él}, 2), (\text{vosotras}, 5), (\text{ella}, 10), (\text{ustedes}, 10), (\text{todos}, 12), (\text{vosotros}, 15), (\text{tú}, 20), (\text{yo}, 23), (\text{ellas}, 30),$

$(\text{ellos}, 25), (\text{nosotros}, 25),$

9) $(\text{ellas}, 30)$ cambia por $(\text{ellos}, 25)$

10) $(\text{ellas}, 30)$ cambia por $(\text{nosotros}, 25)$

finalmente tenemos.

$[(\text{él}, 2), (\text{vosotras}, 5), (\text{ella}, 10), (\text{ustedes}, 10), (\text{todos}, 12), (\text{vosotros}, 15), (\text{tú}, 20), (\text{yo}, 23),$

$(\text{ellos}, 25), (\text{nosotros}, 25), (\text{ellas}, 30)]$

al ejecutar, vemos que ella queda antes que usted y nosotros antes de ellos, lo que pasa ya que al buscar el mínimo de los datos se busca que sea restrictivamente menor, por lo que si tengo que el menor es 5 (elemento antes) y otro con índice mayor también, no lo guardo como menor, sino que solo paso al siguiente índice por lo que el con menor índice queda primero ya que solo lo cambio por el índice inicial (solo nuevo esos datos y los demás quedan donde están)

b) para hacer el selection sort instable podemos hacer que para buscar el menor elemento busque menor o igual, así si hay dos datos repetidos, solo cambiaremos con el inicial al con mayor índice, porque al estar después "smallest" será el con mayor índice, y como i va aumentando con las iteraciones, este se quedará en la posición y no se volverá a cambiar.

Selection Sort (A, n)

for $i = 1, \dots, n-1$

 smallest = i .

 for $j = i+1, \dots, n$.

 if $A[j] \leq A[\text{smallest}]$

 smallest = j .

 intercambiar $A[i]$ con $A[\text{smallest}]$.

por lo que tenemos que sigue ordenando, pero si dos datos son iguales los cambiará de orden.