

Algorithme de tri par sélection

```
def tri_par_selection(t):
    for i in range(len(t)):
        i_min = i
        for k in range(i, len(t)):
            if t[k] < t[i_min] :
                i_min = k
        #for_k_end
        t[i], t[i_min] = t[i_min], t[i]
    #for_i_end
    return t
```

1) effectuer le suivi des variables lors de l'exécution de la commande
 tri_par_selection([70, 40, 10, 30, 20])

=> tableau ci-contre

2) Quelle amélioration proposer pour de la boucle

```
for k in range(i, len(t)):
    Pour éviter la comparaison entre des indices
    indentiques :
    for k in range(i+1, len(t)):
```

3) Quelle amélioration proposer pour de la boucle

```
for i in range(len(t)):
    peut être remplacé par
    for i in range(len(t)-1):
```

t					i	k	i_min	t[k]<t[i_min]
0	1	2	3	4				
[70, 40, 10, 30, 20]					0		0	
[70, 40, 10, 30, 20]						0		t[0]<t[0]:False
[70, 40, 10, 30, 20]						1	1	t[1]<t[0]:True
[70, 40, 10, 30, 20]						2	2	t[2]<t[1]:True
[70, 40, 10, 30, 20]						3		t[3]<t[2]:False
[70, 40, 10, 30, 20]						4		t[4]<t[2]:False
[10, 40, 70, 30, 20]					0		2	
[10, 40, 70, 30, 20]					1	1	1	40<40:False
[10, 40, 70, 30, 20]						2		t[2]<t[1]:False
[10, 40, 70, 30, 20]						3	3	t[3]<t[1]:True
[10, 40, 70, 30, 20]						4	4	t[4]<t[3]:True
[10, 20, 70, 30, 40]					1		4	
[10, 20, 70, 30, 40]					2	2	2	70<70:False
[10, 20, 70, 30, 40]						3	3	30<70:True
[10, 20, 70, 30, 40]						4	3	40<30: False
[10, 20, 30, 70, 40]					2		3	
[10, 20, 30, 70, 40]					3	3	3	70<70:False
[10, 20, 30, 70, 40]						4	4	40<70:True
[10, 20, 30, 40, 70]					3		4	
[10, 20, 30, 40, 70]					4	4	4	70<70:False
[10, 20, 30, 40, 70]								