

```

1 def tri_selection(tab):
2
3     for i in range(len(tab)):
4         0
5         # trouver le min
6         min = i
7         0
8
9         for j in range(i+1, len(tab)):
10            1er passage 1
11            0
12            1
13            5
14            if tab[min] > tab[j]: Vraie
15                0
16                1
17                10
18                7
19
20                min = j
21                1
22                1

```

```

def tri_selection(tab):
    for i in range(len(tab)):
        # trouver le min
        min = i
        for j in range(i+1, len(tab)):
            if tab[min] > tab[j]:
                min = j

        # permuter avec élément indice i
        tmp = tab[i]
        tab[i] = tab[min]
        tab[min] = tmp

    return tab

```

```

1 def tri_selection(tab):
2
3     for i in range(len(tab)):
4         0
5
6         min = 1
7         0
8
9         for j in range(i+1, len(tab)):
10            2ème passage 2
11            0
12            1
13            2
14            if tab[min] > tab[j]: Vraie
15                1
16                2
17                7
18                3
19
20                min = j
21                2
22                2

```

```

print (tri_selection([10,7,3,5,8]))

```

indice	0	1	2	3	4
tab =	10	7	3	5	8
	tab[0]	tab[1]	tab[2]	tab[3]	tab[4]

```

21 # permuter
22 tmp = tab[i]
23
24 tab = [ , , , , ]
25         indice 0 1 2 3 4
26         tab[0] tab[1] tab[2] tab[3] tab[4]
27
28 tab[i] = tab[min]
29
30 tab = [ , , , , ]
31         indice 0 1 2 3 4
32         tab[0] tab[1] tab[2] tab[3] tab[4]
33
34 tab[min] = tmp
35
36 tab = [ , , , , ]
37         indice 0 1 2 3 4
38         tab[0] tab[1] tab[2] tab[3] tab[4]

```

```

1 def tri_selection(tab):
2
3     for i in range(len(tab)):
4         0
5
6         min = i
7         0
8
9         for j in range(i+1, len(tab)):
10            3ème passage 3
11            0
12            1
13            2
14            if tab[min] > tab[j]: FAUX
15                2
16                3
17                3
18                5
19
20                min = j Pas exécuté
21                (min vaut toujours 2)

```

Donner les valeurs des différentes variables du programme (ligne 22 à 30) lors de la 1^{ère} sortie de la boucle interne j (fin du cycle décrit à gauche pour lequel i=0).

Compléter le tableau « tab » ci-dessus après exécution de chacune des instructions

```

1 def tri_selection(tab):
2
3     for i in range(len(tab)):
4         0
5
6         min = i
7         0
8
9         for j in range(i+1, len(tab)):
10            4ème passage 4
11            0
12            1
13            3
14            if tab[min] > tab[j]: FAUX
15                3
16                4
17                3
18                8
19
20                min = j Pas exécuté
21                (min vaut toujours 2)
22
23            On sort de la boucle j

```

```

1 def tri_selection(tab):
2
3     for i in range(len(tab)):
4         1
5         # trouver le min
6         min = i
7
8
9         for j in range(i+1, len(tab)):
10
11
12
13         if tab[min] > tab[j]:
14
15
16
17             min = j
18
19
20

```

On vous aidant de ce qui a été fait précédemment, donner les valeurs des variables à chaque passage dans la boucle j (ligne 9 à 17), Démarrer avec j=2 (car i=1)

Donner les valeurs des différentes variables du programme (ligne 22 à 30) lors de la 2ème sortie de la boucle interne j (fin du cycle décrit à gauche pour lequel i=1) puis compléter le tableau tab en fin de boucle j

Sans tout reprendre à zero, compléter le tableau ci-dessous jusqu'à la fin du programme de tri (pour i=2 puis i=3)

```

1 def tri_selection(tab):
2
3     for i in range(len(tab)):
4         1
5         # trouver le min
6         min = i
7
8
9         for j in range(i+1, len(tab)):
10
11
12
13         if tab[min] > tab[j]:
14
15
16
17             min = j
18
19
20

```

```

21 # permuter
22 tmp = tab[i]
23
24
25
26 tab[i] = tab[min]
27
28
29
30 tab[min] = tmp
31
32

```

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]

```

1 def tri_selection(tab):
2
3     for i in range(len(tab)):
4         1
5         # trouver le min
6         min = i
7
8
9         for j in range(i+1, len(tab)):
10
11
12
13         if tab[min] > tab[j]:
14
15
16
17             min = j
18
19
20

```

```

21 # permuter
22 tmp = tab[i]
23
24
25
26 tab[i] = tab[min]
27
28
29
30 tab[min] = tmp
31
32

```

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]

```

21 # permuter
22 tmp = tab[i]
23
24
25
26 tab[i] = tab[min]
27
28
29
30 tab[min] = tmp
31
32

```

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]

indice 0 1 2 3 4
tab = [, , , ,]
tab[0] tab[1] tab[2] tab[3] tab[4]