

```

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

```

```

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           1
11
12
13            if tab[min] > tab[j]: Vraie
14                1           2
15                7           3
16
17                min = j
18                2           2
19
20

```

| 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
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] |

```

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           1
11
12
13            if tab[min] > tab[j]: FAUX
14                2           3
15                3           5
16
17                min = j Pas exécuté
18                (min vaut toujours 2)
19
20

```

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           1
11
12
13            if tab[min] > tab[j]: FAUX
14                3           4
15                3           8
16
17                min = j Pas exécuté
18                (min vaut toujours 2)
19
20            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]