```
□def tri_selection(tab):
 2 3
           4 5
 6789
               for j in range (i+1 , len(tab)):
10
      1er passage 1
11
13
14
15
16
17
                        min = j
18
                              1
     □def tri_selection(tab):
 2
           for i in range(len(tab)):
    0
 4 5
 6789
                for j in range (i+1 , len(tab)):
10
     2ème passage 2
11
12
13
14
15
16
17
18
19
     □def tri_selection(tab):
 3
           for i in range(len(tab)):
 4 5
 6789
                for j in range (i+1 , len(tab)):
10
    3ème passage 3
12
13
14
15
16
17
                         min = j Pas exécuté
18
                                  (min vaut toujours 2)
19
     □def tri selection(tab):
 3
            for i in range(len(tab)):
 4
 5 6 7
 8
                for j in range (i+1 , len(tab)):
10
    4ème passage 4
12
13
14
15
16
17
                         min = j Pas exécuté
18
                                 (min yaut toujours 2)
                 On sort de la boucle j
```

```
def tri_selection(tab):
    for i in range(len(tab)):
        # trouver le min
        min = i
        for j in range (i+l , 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
```

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

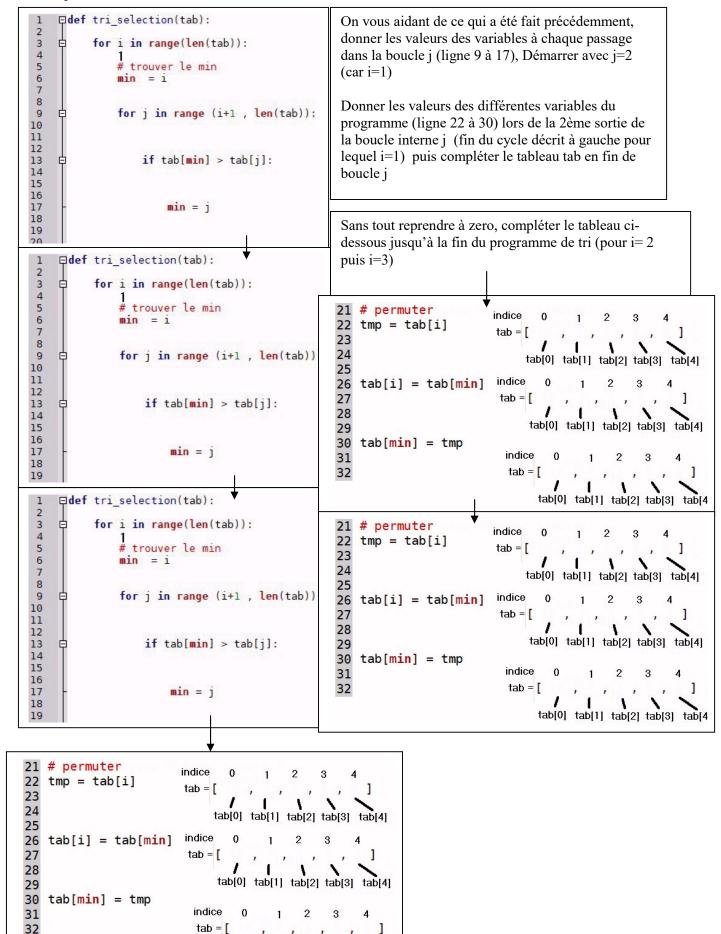
```
# permuter
21
                              indice
22 \text{ tmp} = \text{tab[i]}
                              tab = [
23
24
                                    tab[0] tab[1] tab[2] tab[3] tab[4]
25
                              indice
26
    tab[i] = tab[min]
                                                   2
27
                               tab = [
                                                                  ]
28
                                     tab[0] tab[1] tab[2] tab[3] tab[4]
29
30 \, tab[min] = tmp
31
                                indice
32
                                                                   ]
                                 tab = [
                                      tab[0] tab[1] tab[2] tab[3] tab[4]
```

Donner les valeurs des différentes variables du programme (ligne 22 à 30) lors de la 1<sup>ère</sup> 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

Lycée Louis Payen Page 1 sur 2

Tri par sélection 1<sup>ère</sup> NSI



Lycée Louis Payen Page 2 sur 2

tab[0] tab[1] tab[2] tab[3] tab[4