

Parcours séquentiel - correction

Christophe Viroulaud

Première NSI

Tableau de 30 notes aléatoires

```
1 from random import randint
2
3 les_notes = [randint(0, 20) for _ in range(30)]
```

```
1 def maxi(tab: list ) -> int:
2     """
3     Renvoie la valeur maximale de tab
4     """
5     note_max = 0
6     for note in tab:
7         if note > note_max:
8             note_max = note
9     return note_max
```

Notes minimale et maximale

```
1 def extrema(tab: list ) -> tuple:
2     """
3     Renvoie le mini et le maxi de tab
4
5     Args:
6         tab ( list ):
7     Returns:
8         tuple: (mini, maxi)
9     """
10    mini = 20
11    maxi = 0
12    for note in tab:
13        if note < mini:
14            mini = note
15        if note > maxi:
16            maxi = note
17    return (mini, maxi)
```

Note maximale et sa première position

```
1 def maxi_position(tab: list) -> tuple:
2     """
3     renvoie le max et sa première position dans le
4     tableau
5     """
6     # On peut affecter plusieurs variables sur 1 ligne
7     indice, note_max = 0, 0
8     for i in range(len(tab)):
9         if tab[i] > note_max:
10             note_max = tab[i]
11             indice = i
12     return (indice, note_max)
```

Note maximale et sa dernière position

Recherche
d'extremum

```
1 def maxi_position_dernier(tab: list ) -> tuple:
2     """
3     renvoie le max et sa dernière position dans le
4     tableau
5     """
6     # On peut affecter plusieurs variables sur 1 ligne
7     indice , note_max = 0, 0
8     for i in range(len(tab)):
9         # il suffit juste de modifier la comparaison
10        if tab[i] >= note_max:
11            note_max = tab[i]
12            indice = i
13    return (indice , note_max)
```

Nombre de notes max

```
1 def maxi_nb(tab: list) -> int:
2     """
3     renvoie le nombre d'occurences du maximum
4     """
5     nb, note_max = 0, 0
6     for note in tab:
7         if note > note_max:
8             # ré initialisation du max
9             note_max = note
10            nb = 1
11        elif note == note_max:
12            nb += 1
13    return nb
```