

Enonce

L'ordre des gènes sur un chromosome est représenté par un tableau `ordre` de `n` cases d'entiers distincts deux à deux et compris entre 1 et `n`.

Par exemple, `ordre = [5, 4, 3, 6, 7, 2, 1, 8, 9]` dans le cas `n = 9`.

On dit qu'il y a un point de rupture dans `ordre` dans chacune des situations suivantes :

- la première valeur de `ordre` n'est pas 1 ;
- l'écart entre deux gènes consécutifs n'est pas égal à 1 ;
- la dernière valeur de `ordre` n'est pas `n`.

Par exemple, si `ordre = [5, 4, 3, 6, 7, 2, 1, 8, 9]` avec `n = 9`, on a

- un point de rupture au début car 5 est différent de 1
- un point de rupture entre 3 et 6 (l'écart est de 3)
- un point de rupture entre 7 et 2 (l'écart est de 5)
- un point de rupture entre 1 et 8 (l'écart est de 7)

Il y a donc 4 points de rupture.

Compléter les fonctions Python `est_un_ordre` et `nombre_points_rupture` proposées à la page suivante pour que :

- la fonction `est_un_ordre` renvoie `True` si le tableau passé en paramètre représente bien un ordre de gènes de chromosome et `False` sinon ;
- la fonction `nombre_points_rupture` renvoie le nombre de points de rupture d'un tableau passé en paramètre représentant l'ordre de gènes d'un chromosome.

Script Python

```

1 def est_un_ordre(tab):
2     """
3         Renvoie True si tab est de longueur n et contient tous les entiers
4         de 1 à n, False sinon
5     """
6     for i in range(1,...):
7         if ...:
8             return False
9     return True
10
11
12 def nombre_points_rupture(ordre):
13     """
14         Renvoie le nombre de point de rupture de ordre qui représente un ordre
15         de gènes de chromosome
16     """

```

```

17     assert ... # ordre n'est pas un ordre de gènes
18     n = len(ordre)
19     nb = 0
20     if ordre[...] != 1: # le premier n'est pas 1
21         nb = nb + 1
22     i = 0
23     while i < ...:
24         if ... not in [-1, 1]: # l'écart n'est pas 1
25             nb = nb + 1
26             i = i + 1
27     if ordre[...] != n: # le dernier n'est pas n
28         nb = nb + 1
29     return nb

```

Exemples :

Script Python

```

>>> est_un_ordre([1, 6, 2, 8, 3, 7])
False
>>> est_un_ordre([5, 4, 3, 6, 7, 2, 1, 8, 9])
True
>>> nombre_points_rupture([5, 4, 3, 6, 7, 2, 1, 8, 9])
4
>>> nombre_points_rupture([1, 2, 3, 4, 5])
0
>>> nombre_points_rupture([1, 6, 2, 8, 3, 7, 4, 5])
7
>>> nombre_points_rupture([2, 1, 3, 4])
2

```