

Remarques DS 4 (info)

Les notes d'informatique sont très hétérogènes. La moyenne est de 4,2/10 avec un écart type de 2. Le barème était le suivant :

- question 1 : 1 point
- question 2 : 2,5 points
- question 3 : 2,5 points
- question 4 : 2 points
- question 5 : 2 points

Les points ci-après sont mal réussis dans l'ensemble et méritent d'être réexpliqués. Je rappelle par ailleurs qu'une erreur d'indentation est signalée par `IND`, et que l'abréviation `Qx` signale que Python indiquerait le message d'erreur suivant : `name error : name 'x' is not defined`.

Remarques générales :

- `RQP` : Apprenez à réutiliser les questions précédentes, c'était par exemple ce qui était attendu à la question 5, très peu réussie, et qu'il faut revoir.

Réutiliser une fonction implique de ne pas recopier les mêmes calculs, mais de faire un appel à une fonction. Ainsi, pour la fonction `moy_bis`, on s'attendait à lire quelque chose du type `moy(Lp)` et non pas `sum(Lp)/len(Lp)`, ce calcul étant déjà été explicité à la question 1.

Notez enfin que pour réutiliser le résultat d'une fonction, il faut lui donner un nom. Ainsi pour la question 5, parmi les codes ci-dessous, celui de gauche est correct, mais celui de droite ne l'est pas.

```
1 def moy_bis(L) :
2     Lp,Lm = separe(L)
3     # puis instructions
4     utilisant Lp et Lm
```

```
1 def moy_bis(L) :
2     separe(L)
3     # puis instructions utilisant
        des variables Lp et Lm
        non introduites donc n'
        ayant pas de sens
```

- `IPM` : Vous ne pouvez pas utiliser les symboles $+$ et $-$ dans les noms de variables, même en les plaçant en indices ou avec un tiret. En effet, pour Python, l'instruction `L+=[]` ne signifie pas qu'il faut définir une variable `L+` contenant `[]`, mais correspond à `L = L+[]`, c'est-à-dire à concaténer `[]` à la liste `L`. Notez par ailleurs que, même à l'écrit, votre code Python doit pouvoir être tapé sur un clavier sans ajustement. Ainsi, vous pouvez pas choisir de nommer une variable `L1`, mais vous pouvez l'appeler `L1` ou `L_1`.
- Encore trop de copies recopient sans se poser de question le schéma suivant dès qu'il faut faire un graphique :

```
1 a,b,N = ...
2 L_absi = [a + k*(b-a)/N for k in range(N+1)]
3 L_ordo = [f(x) for x in L_absi]
```

C'est effectivement ce qu'il faut faire pour tracer le graphe d'une fonction de la variable réelle $f : [a, b] \rightarrow \mathbb{R}$. En revanche, c'est tout à fait inadapté dans beaucoup de situations... en particulier quand les valeurs à mettre en abscisse sont simplement des entiers, ou quand il n'y a pas de fonction f ! Si vous avez commis cette erreur, revoir par exemple les exercices 7 et 8 de la feuille de cours 6 d'informatique, ou l'interro 13.

- Sauts de page interdits en Python : l'indentation étant essentielle en Python, vous ne pouvez pas vous permettre de changer de page au sein d'une fonction.