TP 13 - Révisions

Exercice 1. Conditions

Compléter le code ci-dessous de manière à ce qu'à la fin `a`, contiennent la plus grande valeur, `b` celle du milieu et `c` la plus petite valeur.

```
1  a=int(input())
2  b=int(input())
3  c=int(input())
4  # A faire
```

Exercice 2. Boucles

1. Écrire une fonction `factorielle` qui prend un entier `n` et renvoie `n` factorielle.
On rappelle que $n!$ vaut $1 \times 2 \times 3 \cdots \times n$ pour tout $n \geq 1$ et que $0! = 1$.
2. Écrire une fonction `demande` qui demande un nombre à l'utilisateur jusqu'à ce qu'il soit compris entre 0 et 20 et le renvoie.
3. Écrire une fonction `listeCroissante` qui laisse l'utilisateur rentrer autant de nombres positifs qu'il veut, tant que chaque nombre est strictement supérieur à celui d'avant. Elle renvoie ensuite la liste des nombres entrés sans le dernier.

Exercice 3. Listes

1. Écrire une fonction `maximum` qui prend une liste d'entiers non nécessairement positifs et renvoie un tuple avec son maximum et la liste des indices où il apparait.
Vous ne pourrez pas utiliser la fonction `max` de Python.
2. Écrire une fonction `sansDoublons` qui prend une liste d'entiers et renvoie la liste contenant les mêmes nombres mais au plus une fois et dans le même ordre.
Par exemple, `singletons([2,1,2,1,3,2,1,4])` renverra la liste `[2,1,3,4]`.

Exercice 4. Chaînes

1. Écrire une fonction `sansPonctuation` qui prend une chaîne et renvoie la chaîne dans laquelle on a supprimé tous les symboles de ponctuation.
2. Écrire une fonction `nbMots` qui prend une chaîne et renvoie son nombre de mots.
3. Écrire une fonction `replace` qui prend une chaîne et un dictionnaire et qui renvoie la chaîne obtenue en remplaçant chaque clé du dictionnaire par sa valeur.
Par exemple, `replace("Bien le bonjour madame",{'madame' : 'monsieur', 'bonjour' : 'aurevoir'})` renverra "Bien le aurevoir monsieur".

Exercice 5. Dictionnaires

1. Écrire une fonction `freqChaine` qui prend un chaîne de caractères et renvoie un dictionnaire qui contient pour chaque lettre de la chaîne son nombre d'occurrences.
Par exemple, `freqChaine('abbabba!')` renverra `{ 'a' : 3 , 'b' : 4 , '!' : 1}`.
2. Écrire une fonction `copie` qui renvoie une copie d'un dictionnaire de listes.