

Système - TP de familiarisation avec Python - correction Pierre Pompidor

Tous les scripts suivants pourraient commencer par l'entête :
(la première ligne n'ayant un sens que pour Bash)

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
```

Ecrivez un script Python qui affiche "Bonjour"

```
print("Bonjour")
```

Ecrivez un script qui vérifie qu'un paramètre lui a bien été passé
et dans ce cas l'affiche après "Bonjour" :

```
import sys
if len(sys.argv) == 1 :
    print("Vous devez donner un paramètre au script")
else :
    print("Bonjour", sys.argv[1])
```

Affichez tous les paramètres du script :

```
import sys
for parametre in sys.argv :
    print(parametre)
```

Affichez tous les paramètres du script sauf le premier :

```
import sys
for parametre in sys.argv[1:] : # utilisation d'un slice
    print(parametre)
```

Écrivez un script qui calcule en itératif la factorielle de n :

```
import sys

if len(sys.argv) == 1 :
    n = int(sys.argv[1]) # la fonction int() transtype le paramètre en entier
    r = 1
    while n > 1 :
        r = r * n # ou r *= n
        n = n - 1 # ou n -= 1
        print("Résultat =", r)
else :
    print("Vous devez donner un paramètre au script")
```

Écrivez un script qui affiche les nombres premiers contenus dans les n premiers entiers positifs, avec réutilisation des nombres premiers déjà déterminés

```
import sys

n = int(sys.argv[1])
listeNP = [2]
candidat = 3
while candidat <= n :
    booleenNP = True # on considère que le candidat est premier
    for i in listeNP :
        if candidat % i == 0 : # calcul d'un modulo (reste de la division entière)
            booleenNP = False
            break # Pour arrêter la boucle la plus proche (*)
    if booleenNP == True :
        listeNP.append(candidat)
    candidat = candidat + 1

print(listeNP)
```

() l'utilisation du break peut-être critiquée, que faudrait-il faire sinon pour arrêter la boucle de manière plus "propre" (mais sans doute aussi de manière moins efficace) ?*