



Institut Supérieur d'Informatique et Mathématiques de Monastir

Département d'Informatique

SECTION : LICENCE EN INFORMATIQUE,

NIVEAU : 1^{ère} ANNÉE , A.U : 2019-2020

Matière : Programmation Python

Travaux Pratiques N°6

Exercice 1

Un nombre est dit super premier s'il est premier et si en supprimant des chiffres à partir de sa droite, le nombre restant est aussi premier.

Exemple :

Le nombre 59399 est super premier car les nombre 59399, 5939, 593, 59 et 5 sont tous premier.

Écrire un programme Python (en utilisant les modules) qui permet de :

- Remplir une liste L par N ($5 \leq N \leq 50$) entiers positifs,
- déterminer et afficher tous les entiers super premiers de la liste.

Exemple :

Soit la liste L suivante :

$L = \{370, 59399, 272, 100, 797, 379, 119, 2393\}$

Résultat :

Les nombres super premiers de la liste sont

59399 797 379 2393

Exercice 2

Un entier N est dit premier-factoriel s'il vérifie les deux propriétés suivantes :

- N est premier
- N s'écrit sous la forme d'une factorielle incrémenté ou décrétementé de 1 ($N=F! - 1$ ou $N=F! + 1$).

Exemples :

- 7 est premier-factoriel car 7 est premier et $7 = 3!+1$; $3!=3*2*1$
- 23 est premier-factoriel car 23 est premier et $23 = 4!-1$; $4!=4*3*2*1$
- 17 n'est pas premier-factoriel car $17-1=16$ n'est pas factorielle et $17+1=18$ n'est pas aussi une factorielle

Écrire un programme Python (en utilisant les modules) qui permet de :

- Remplir une liste L par N ($5 \leq N \leq 50$) entiers positifs,
- déterminer et afficher tous les entiers premiers-factoriel de la liste.

Exemple :

Soit la liste L suivante :

$L = \{23, 2000, 5039, 100, 719, 70, 17, 2393\}$

Résultat :

Les entiers premiers-factoriel de la liste sont

23 5039 719 17

Exercice 3

Un nombre de smith s'il est un nombre dont la somme des chiffres est égale à la somme de tous les chiffres de ses facteurs premiers.

Exemple : Si $X = 636$

Les facteurs premiers de 636 sont 2, 2, 3, 53

La somme des chiffres de 636 est : 15

La somme des chiffres de ses facteurs premiers est $2+2+3+5+3=15$

Donc 636 est un nombre smith

Écrire un programme Python (en utilisant les modules) qui permet de :

- Remplir une liste L par N ($5 \leq N \leq 50$) entiers positifs,
- déterminer et afficher tous les entiers smith de la liste.

Exemple :

Soit la liste L suivante :

$L = \{23, 2000, 29, 100, 13, 70, 27, 636\}$

Résultat :

Les nombres smith de la liste sont

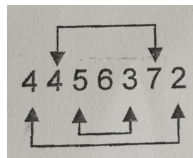
29 13 27 636

Exercice 4

Écrire un programme Python (en utilisant les modules) qui permet de :

- Remplir une liste L par N ($5 \leq N \leq 50$) entiers naturels Anti palindrome ,
- Remplacer chaque entier X du tableau par le plus grand nombre qu'on peut forme par les chiffres de X sans répétition.

Un entier naturel est dit Anti palindrome, si les chiffres corespondants à partir de la droite et de la gauche sont **différents deux à deux**. Le nombre suivant est Anti palindrome



Exemple :

Pour N = 5

$L = \{24526, 522316, 453675, 66328, 723254\}$

$L = \{65422, 653221, 765543, 86632, 754322\}$

$L = \{6542, 65321, 76543, 8632, 75432\}$

Résultat :

6542 65321 76543 8632 75432

Exercice 5

La suite U_n est définie comme suit :

$$U_0 = X \quad (X \text{ un entier naturel de 4 chiffres})$$

$$U_n = \text{Max}(U_{n-1}) - \text{Min}(U_{n-1})$$

Pour calculer terme de la suite :

- On détermine le plus grand et le plus petit entier naturel qu'on peut former par les chiffres du terme précédent.

- On calcul $U_i = \text{Max}(U_{i-1}) - \text{Min}(U_{i-1})$
- On refait les étapes précédentes pour calculer les termes de la suite
- On arrête lorsque la suite devient stationnaire ($U_i = U_{i-1}$)

Travail demandé :

Ecrire un programme Python(en utilisant les module) qui permet de saisir le premier terme U_0 de calculer et d'afficher les termes de cette suite ainsi que leur nombre.

Exemple :

$U_i = 9324$

Résultats :

9324 7083 8352 6174

Le nombre des termes de la suite est : 4

Exercice 6

Pour déterminer si un nombre formé d'un grand nombre de chiffres est divisible par 13, il suffit de :

- Saisir le nombre N
- Séparer ce nombre par tranche de 3 chiffres en partant des unités
- Remplir une liste L par les tranches découpées
- Calculer la valeur retournée V par la liste en insérant alternativement des - et des + entre les tranches à partir du début du nombre en commençant par un -
- Vérifier si la valeur absolue du nombre ainsi obtenu est multiple de 13

Exemple :

Soit le nombre $N = 1633123612311854$

on le sépare par tranche de trois à partir des unités, on obtient le tableau suivant :

$L = \{854, 311, 612, 123, 633, 1\}$

La valeur calculée $V = 1 - 633 + 123 - 612 + 311 - 854 = -1664$

1664 est divisible par 13 donc 1633123612311854 est divisible par 13