

DEVOIR SEMESTRE 2

Matière : INFORMATIQUE

Classe : 1^{ère} Année GB

Durée : 1h

*Toutes les réponses doivent être rédigées en Python.
L'indentation dans les instructions Python sera prise en considération dans le barème*

Exercice 1 : (10 points)

Un entier naturel N est dit sympathique si tous ses chiffres sont distincts (pas de chiffres qui se répètent) et s'il est multiple de la somme de ses chiffres.

Exemple : $N=24$ est un nombre sympathique car les chiffres de N sont distincts et $24 \bmod (2+4)=0$.

1. Définir la fonction **saisie()** qui permet de saisir et retourner un entier strictement positif.
2. Définir la fonction **somme(N)** qui retourne la somme des chiffres du paramètre N .
3. Définir la fonction **distinct(N)** qui retourne **True** si tous les chiffres du paramètre N sont distincts et **False** sinon.
4. Définir la fonction **sympathique(N)** qui retourne **True** si le paramètre N est sympathique et **False** sinon.
5. Ecrire un programme principal permettant de :
 - Saisir un entier positif X .
 - Vérifier si X est un nombre sympathique ou non.
 - Créer et afficher une **liste L** contenant tous les nombres sympathiques compris entre 10 et 1000.

Exercice 2 : (10 points)

Dans le cadre d'un programme de santé et de bien-être, une application informatique est nécessaire pour aider les utilisateurs à évaluer leur poids et à suivre leur progression vers un poids idéal pour une bonne santé. Il s'agit, dans ce cas, de concevoir un système informatique qui permettra aux utilisateurs de :

- Entrer leurs données personnelles telles que le sexe, le poids et la taille.
 - Enregistrer leur poids actuel et suivre les fluctuations de poids au fil du temps.
 - Proposer des recommandations personnalisées pour atteindre un poids idéal en fonction des objectifs de santé individuels.
1. Définir la fonction **sauvegarde (n)** qui permet de sauvegarder n mesures, saisies au clavier, d'une personne dans un fichier texte nommé '**mesures.txt**'.
 - ✓ La première ligne du fichier contient le sexe de la personne saisi comme une chaîne de caractères ayant la valeur 'F' (Féminin) ou 'M' (Masculin).

- ✓ Pour les autres (n-1) lignes, chaque ligne contient, dans l'ordre, les trois données suivantes séparées par le caractère ';' :
- **Annee** : un entier qui représente une année entre 1900 et 2024.
- **Taille** : un réel qui représente la taille normale en cm entre 120 et 200.
- **Poids** : un réel qui représente le poids en kg entre 25 et 150.

Exemple : voici un exemple du fichier 'mesures.txt' contenant les lignes suivantes :

```
M
2015 ;159 ;42.5
2016 ;160 ;45.0
2017 ;162 ;50.7
2018 ;168 ;61.0
```

2. Définir la fonction **poidsParfait(T,S)** qui retourne le poids parfait d'une personne (en kg) selon sa taille T(en centimètre) et son sexe S (une chaîne de caractères ayant la valeur 'F' (Féminin) ou 'M' (Masculin)) calculé par les deux formules suivantes:
 - **Poids parfait Masculin** = $Taille - 100 - ((Taille - 150) / 4)$
 - **Poids parfait féminin** = $Taille - 100 - ((Taille - 150) / 2.5)$
3. Définir la fonction **analyseDonnees()** qui, à partir du fichier 'mesures.txt', lit les mesures pour retourner une **liste de tuples**. Chaque tuple contient deux éléments :
 - ✓ Le premier élément est le résultat de comparaison entre le poids réel et le poids parfait de la personne noté par :
 - **'Inf'** : si le poids en kg de la personne est inférieur à son poids parfait
 - **'Sup'** : si le poids en kg de la personne est supérieur à son poids parfait
 - **'Egal'** : si le poids en kg de la personne est égal à son poids parfait
 - ✓ Le deuxième élément est la différence entre les deux poids.

Pour l'exemple donné dans la question 1, l'appel de la fonction **analyseDonnees()** retourne [(' Inf',14.25), (' Inf',12.5), (' Inf',8.3), (' Inf',2.5)]