

## EXERCICE 5 (4 points)

*Principaux thèmes abordés : Traitement de données en tables (CSV) et langages et programmation (spécification).*

Afin d'améliorer l'ergonomie d'un logiciel de traitement des inscriptions dans une université, un programmeur souhaite exploiter l'intelligence artificielle pour renseigner certains champs par auto-complétion. Il s'intéresse au descripteur « genre » (masculin, féminin ou indéterminé). Pour cela il propose d'exploiter les dernières lettres du prénom pour proposer automatiquement le genre.

Pour vérifier son hypothèse, il récupère un fichier CSV associant plus de 60 000 prénoms du monde entier au genre de la personne portant ce prénom. En utilisant seulement la dernière lettre, le taux de réussite de sa démarche est de 72,9% avec la fonction définie ci-dessous :

```
1 def genre(prenom):
2     liste_M = ['f', 'd', 'c', 'b', 'o', 'n', 'm', 'l', 'k',
3               'j', 'é', 'h', 'w', 'v', 'u', 't', 's', 'r',
4               'q', 'p', 'i', 'p', 'z', 'x', 'ç', 'ö', 'ã',
5               'â', 'ï', 'g']
6     liste_F = ['e', 'a', 'ä', 'ü', 'y', 'ë']
7
8     if prenom[len(prenom)-1].lower() in liste_M :
9         return "M"
10    elif prenom[len(prenom)-1].lower() in liste_F :
11        return "F"
12    else :
13        return "I"

# Pour rappel, C.lower() convertit le caractère C en minuscule.
```

### 1. Appropriation

- Expliquer ce qu'est un fichier CSV.
- Donner le type de l'argument `prenom` de la fonction `genre`, et le type de la réponse renvoyée.

## 2. Développement

Pour effectuer son étude sur les prénoms à partir du fichier CSV, le programmeur souhaite utiliser la bibliothèque csv.

a) La bibliothèque csv est un module natif du moteur python.

Donner, dans ce cas, l'instruction d'importation.

b) Au cours du développement de son projet, le programmeur souhaite insérer une assertion sur l'argument donné à la fonction.

Proposer une assertion sur le type de l'argument qui corrige une erreur lorsque le type ne correspond pas à celui attendu.

c) Avant le déploiement de sa solution, le programmeur décide de rendre sa fonction plus robuste.

Pour cela il veut remplacer l'assertion proposée dans la question **2.b)** par une gestion de l'argument pour éviter toutes erreurs empêchant la poursuite du programme.

Proposer alors une ou plusieurs instructions en Python utilisant l'argument afin de s'assurer que la fonction se termine quel que soit le type de l'argument.

## 3. Améliorations

En prenant en compte les deux dernières lettres du prénom, il parvient à augmenter son taux de réussite à 74,4%. Pour cela, son étude du fichier CSV lui permet de créer deux listes : `liste_M2` pour les terminaisons de deux lettres associées aux prénoms masculins et `liste_F2` pour les prénoms féminins.

Sur votre copie, recopier et modifier la structure conditionnelle (lignes 8 à 13) de la fonction `genre` afin de prendre en compte les terminaisons de deux lettres des listes `liste_M2` et `liste_F2`.