

## Mini-projet : “Adopte un GM”

Ce projet se réalisera seul ou à deux <sup>1</sup>.

### Sujet

Nous allons réaliser plusieurs programmes en python permettant de trouver l’âme sœur.

Un premier ensemble de programmes permettra de gérer les profils des personnes :

- Saisie du profil
- Affichage d’un profil
- Suppression d’un profil

Un second ensemble de programmes vérifiera la compatibilité entre les personnes :

- Comparaison des deux profils et attribution d’un score de compatibilité
- Comparaison entre un profil et tous les autres

### Gestion des profils

Les paramètres liés à une personne seront au minimum (vous pourrez en ajouter selon votre imagination) :

- L’identifiant (pseudo)
- Le genre, sous forme numérique :

1. HOMME

2. FEMME

3. AUTRE

- La date de naissance, sous la forme JJ/MM/YYYY
- Le signe astrologique, sous forme numérique :

1. BELIER

5. LION

9. SAGITAIRE

2. TAUREAU

6. VIERGE

10. CAPRICORNE

3. GEMEAUX

7. BALANCE

11. VERSEAU

4. CANCER

8. SCORPION

12. POISSON

---

1. Un logiciel de détection de plagiat sera utilisé pour comparer les différents projets rendus

- La taille, sous forme numérique :
 

1. < 1,50 m	3. entre 1,60m et 1,70m	5. entre 1,80m et 1,90m
2. entre 1,50m et 1,60m	4. entre 1,70m et 1,80m	6. > 1,90m
- La couleur des cheveux, sous forme numérique :
 

1. Noirs	3. Blonds	5. Gris
2. Châtains	4. Roux	6. Blancs
- La couleur des yeux, sous forme numérique :
 

1. Noirs	3. Verts	5. Gris
2. Marrons	4. Bleus	
- Le genre de la personne recherchée par l'utilisateur
 

1. HOMME	2. FEMME	3. AUTRE	4. PEU_IMPORTE
----------	----------	----------	----------------
- L'âge, la taille, la couleur des cheveux, la couleur des yeux de la personne recherchée (mêmes valeurs possibles que ci-dessus)
- La réponse à 10 questions, sur une échelle de 0 (je déteste) à 5 (j'adore). Des exemples de questions (vous pourrez les modifier ou en ajouter à votre convenance) :
 

– Aimez-vous les animaux ?	– Aimez-vous les jeux de société ?
– Aimez-vous les voyages ?	– Aimez-vous les enfants ?
– Aimez-vous Emmanuel Macron ?	– Aimez-vous programmer en python ?
– Aimez-vous Game of Thrones ?	– Aimez-vous la viande ?
– Aimez-vous cuisiner ?	– Aimez-vous l'argent ?

## Représentation et stockage des profils

Tous les profils seront stockés dans un fichier texte contenant une ligne par personne inscrite. Chaque ligne contiendra tous les paramètres liés à cette personne (valeurs numériques), séparés par des virgules.

Par exemple, la ligne :

martin92,1,28/12/1997,8,4,1,2,2,3,2,3,4,1,2,4,3,5,4,2,1,5

correspondra à un homme né le 24 octobre 1997, de signe scorpion, mesurant entre 1,70m et 1,80m, brun aux yeux marrons, recherchant une femme entre 1,60m et 1,70m, châtain aux yeux verts, ayant répondu respectivement 4, 1, 2, 4, 3, 5, 4, 2, 1, 5 aux 10 questions posées.

Attention, il est impossible que deux personnes aient le même pseudo !

## Fonctionnalités à réaliser

1. Saisie d'un profil : un nouvel utilisateur doit pouvoir saisir son profil en répondant successivement aux questions permettant de bien cerner sa personne et ses attentes.  
**Entrée :** Le fichier des profils  
**Sortie :** Le même fichier, enrichi d'une nouvelle ligne
2. Affichage d'un profil : un visiteur doit pouvoir afficher le profil de tout utilisateur.  
**Entrée :** Le fichier des profils, un nom d'utilisateur  
**Sortie :** Les informations (compréhensibles par un humain) sur cet utilisateur
3. Suppression d'un profil : un utilisateur doit pouvoir supprimer son profil.  
**Entrée :** Le fichier des profils, un nom d'utilisateur  
**Sortie :** Le même fichier, amputé de la ligne concernant l'utilisateur spécifié.

## Appariement des profils

Nous allons comparer les profils en fonction d'une mesure d'appariement. Nous proposons la démarche suivante :

- Deux profils ne peuvent pas s'apparier si les genres recherchés par les deux personnes ne sont pas compatibles.
- Pour les autres paramètres, nous allons implémenter une mesure de distance entre les profils. Plus la distance sera faible, plus les personnes seront promises à un bel avenir. Nous allons pour cela créer des vecteurs de valeurs numériques correspondant à :
  - La différence entre l'âge recherché et l'âge réel, de part et d'autre ;
  - La compatibilité astrologique (voir plus loin des éléments permettant de la calculer à partir du signe astrologique des personnes)
  - La différence entre les "chemins de vie" des deux personnes (voir plus loin comment calculer le chemin de vie d'une personne en numérologie, à partir de sa date de naissance)
  - La différence entre la taille, couleur des cheveux, couleur des yeux réelles et souhaitée de part et d'autre
  - La différence entre les valeurs données en réponse aux 10 questions par les deux personnes.

La distance entre deux personnes sera alors la somme des distances entre chaque composante du vecteur (valeur absolue de la différence). Par exemple, si les réponses respectives à la question 8 sont 2 et 5, alors la distance sur cette composante est 3.

## Détermination de la compatibilité astrologique

Une table de compatibilité des signes astrologiques vous est fournie à la figure 1. Notez bien que dans ce tableau, une valeur élevée correspond à une compatibilité élevée. Il vous faudra donc l'adapter. Réfléchissez également bien à une structure de données efficace pour stocker et utiliser ces informations.

## Détermination du chemin de vie

Pour calculer le chemin de vie d'une personne à partir de sa date de naissance, il suffit d'effectuer les opérations suivantes :

1. Faire la somme des jour, mois et année de naissance de la personne. Par exemple, pour **martin92**, né le 28/12/1997, on obtiendra  $28 + 12 + 1997 = 2037$ .
2. Faire la somme des chiffres du résultat, et répéter l'opération jusqu'à ce que cette somme soit comprise entre 1 et 9. Ainsi, pour **martin92**, on obtiendra  $2 + 0 + 3 + 7 = 12$ , puis au final  $1 + 2 = 3$ .

## Fonctionnalités à réaliser

1. Comparaison entre deux profils.  
**Entrée :** Le fichier des profils, un nom d'utilisateur  
**Sortie :** Une valeur de compatibilité
2. Comparaison entre un profil et tous les autres  
**Entrée :** Le fichier des profils, un nom d'utilisateur  
**Sortie :** Les 5 profils les plus compatibles (ou moins si on n'en trouve pas 5), avec leurs valeurs de compatibilité

Hommes >>> Femmes	Bélier	Taureau	Gémeau	Cancer	Lion	Vierge	Balance	Scorpion	Sagit	Capricor	Verseau	Poisson
Bélier	50	70	50	50	60	50	70	60	70	45	90	45
Taureau	85	90	70	80	55	80	70	90	80	80	60	90
Gémeaux	50	60	75	60	50	60	70	60	70	90	90	40
Cancer	50	95	65	55	95	90	75	80	55	55	70	70
Lion	50	55	45	95	45	70	70	75	90	80	70	45
Vierge	40	55	55	90	75	60	60	80	80	60	40	55
Balance	60	55	65	75	90	60	70	65	90	50	90	55
Scorpion	55	85	60	70	90	70	55	40	95	55	50	90
Sagittaire	60	50	70	60	90	55	85	95	90	65	90	90
Capricorne	60	95	70	65	90	50	45	65	40	85	80	90
Verseau	70	55	80	60	80	40	90	50	75	70	75	70
Poissons	45	90	40	70	50	65	70	65	80	70	45	75

FIGURE 1 – Matrice des compatibilités astrologiques

## Rendu du projet

Le projet est à rendre sur votre espace **moodle**. Il sera composé de :

1. Un ou plusieurs fichiers python, ou un ou plusieurs notebook Jupyter.
2. Un fichier `README.txt` donnant la liste des programmes et comment les utiliser en pratique. Ce fichier doit contenir toutes les instructions nécessaires à l'exécution ; il est très important pour expliquer à l'utilisateur comment se servir de l'outil.
3. Un rapport **au format pdf** de quelques pages décrivant votre projet, les principaux choix effectués (structures de données, algorithmes, interaction avec l'utilisateur), difficultés rencontrées.

La notation tiendra compte des points suivants :

1. Implémentation des fonctionnalités demandées
2. Facilité d'utilisation
3. Clarté du code rendu (commentaires notamment)
4. Qualité du code et des algorithmes (choix des bonnes structures de données, algorithmes simples et efficaces...)