

Travail pratique #2

Walid Boulabiar

Date limite de remise : le dimanche 25 juin 2023 à minuit

TP2 noté sur 30 et compte pour 10% de la note finale

Travail individuel

1 Objectifs

Ce travail a comme principaux objectifs de vous aider à assimiler les concepts vus en cours: l'utilisation des bonnes structures de données, la décomposition fonctionnelle, l'utilisation des fonctions et des modules.

Pensez à écrire du code réutilisable en utilisant les fonctions et modules. Votre défi est de bien décomposer le code.

2 Spécifications des programmes : Le nom de chaque fichier .py doit être le même que celui donné entre parenthèse pour chaque question.

- A.** (chercher_chaine_saisie) Trouver le nombre d'occurrences d'une chaîne dans une autre. Votre programme doit demander à l'utilisateur de saisir deux chaînes de caractères puis il doit compter le nombre de fois que la 2^{ème} chaîne se retrouve dans la 1^{ère} chaîne. Afficher les deux chaînes saisies et la réponse (le nombre d'occurrences).

Exemple 1 :

Chaîne 1 saisie : « Un bon chasseur doit savoir chasser sans son chien »

Chaîne 2 saisie : « ch »

Réponse : La chaîne 2 se retrouve 3 fois dans la chaîne 1.

Exemple 2 :

Chaîne 1 saisie : « bbbbbbbb »

Chaîne 2 saisie : « bb »

Réponse : La chaîne 2 se retrouve 4 fois dans la chaîne 1.

Dans l'exemple 2, nous voyons clairement que le résultat est 4 et non 7 ce qui vous permettra de deviner l'algorithme que vous devez entreprendre.

Contraintes à respecter :

- La 1^{ère} chaîne saisie doit être d'une longueur minimum de 4 caractères.
 - La 2^{ème} chaîne saisie doit être d'une longueur minimale d'un caractère et plus courte que la première.
 - Vous n'avez pas le droit d'utiliser la fonction count implémentée dans Python depuis la classe String.
- B.** (chercher_chaine_fichier) On reprend le même traitement que celui de l'exercice précédent sauf que votre programme doit demander à l'utilisateur de saisir deux chaînes puis, votre programme, doit les stocker dans un fichier texte sous le nom texte.txt. Par la suite, votre programme doit lire le contenu du fichier (1^{ère} ligne représente la 1^{ère} chaîne et la 2^{ème} ligne représente la 2^{ème} chaîne) et faire le traitement pour afficher à l'écran les 2 chaînes ainsi que le nombre d'occurrences de la deuxième chaîne dans la première. Les contraintes de l'exercice précédent doivent être respectées. Pensez à réutiliser le code en appelant la fonction de la question 1 qui compte le nombre d'occurrence.
- C.** (afficher_motif_liste) Créer un programme qui permet d'afficher un menu textuel en mode console qui permet l'affichage d'un motif à base d'étoiles et espaces à partir d'un choix de motif fait par l'utilisateur. Les 4 choix de motifs sont les suivants : (voir les captures d'écrans à partir de la page 4 pour mieux comprendre le résultat demandé) :
- 'F' pour la forme de Fenêtre
 - 'L' pour la forme de Losange
 - 'C' pour la forme de Coin
 - Et 'X' pour la forme de X

Contraintes à respecter :

- Il faut afficher un menu en mode console pour faire la saisie de la dimension puis la lettre du motif.

- Définir une **liste** (de 2 dimensions) à partir d'une dimension saisie par l'utilisateur et le remplir du motif choisi. La dimension saisie doit être une valeur **impaire** entre **5 et 39 inclusivement**.
Attention : la construction du motif doit être dans une liste et non une chaîne de caractères.
- La saisie de la dimension doit être validée à l'utilisateur avec rétroaction (feedback). Ceci dit, si la dimension n'est pas valide, votre programme doit demander de nouveau à l'utilisateur, une dimension, jusqu'à l'obtention d'une valeur valide.
- La saisie de la lettre doit être validée à l'utilisateur avec rétroaction (feedback). Ceci dit, les seules lettres acceptées sont : (F, L, C et X) en minuscule ou majuscule, sinon, pour toute autre lettre, votre programme doit demander de nouveau à l'utilisateur, une lettre, jusqu'à l'obtention d'une lettre valide.
- Après la saisie des valeurs « dimension et lettre », vous devez remplir la liste, préalablement instancié, par le motif choisi. L'utilisation de boucles et de structures conditionnelles seront la base de vos algorithmes pour créer le motif voulu dans la liste.
- Après l'affichage du motif, il doit être possible de redonner une dimension et refaire un choix de motif autant de fois que l'on veut.

Voici des imprimés écrans de différents cas d'utilisations :

```

Saisir la dimension de la liste (entre 5 et 39) ?
Choisir une lettre identifiant le motif voulu?

F : Fenetre
L : Losange
C : Coin
X : Le X

F
*****
* * *
* * *
*****
* * *
* * *
*****

Voulez-vous encore faire un choix de motif O/N?

```

```

L : Losange
C : Coin
X : Le X

F
*****
* * *
* * *
*****
* * *
* * *
*****

Voulez-vous encore faire un choix de motif O/N?
Saisir la dimension de la liste (entre 5 et 39) ?
Choisir une lettre identifiant le motif voulu?

F : Fenetre
L : Losange
C : Coin
X : Le X

L
*****
* * *
* * *
*****
* * *
* * *
*****

Voulez-vous encore faire un choix de motif O/N?

```

```
Voulez-vous encore faire un choix de motif O/N?o
Saisir la dimension de la liste (entre 5 et 39)9
Choisir une lettre identifiant le motif voulu?
```

```
F : Fenetre
L : Losange
C : Coin
X : Le X
```

```
C
  x x x x
  x x x
  x x
  x
```

```
      x
      x x
      x x x
      x x x x
```

```
Voulez-vous encore faire un choix de motif O/N?
```

```
Voulez-vous encore faire un choix de motif O/N?o
Saisir la dimension de la liste (entre 5 et 39)11
Choisir une lettre identifiant le motif voulu?
```

```
F : Fenetre
L : Losange
C : Coin
X : Le X
```

```
X
  *          *
  *          *
  *          *
  *          *
  *          *
  *          *
  *          *
  *          *
  *          *
  *          *
  *          *
```

```
Voulez-vous encore faire un choix de motif O/N?
```

```
Saisir la dimension de la liste (entre 5 et 39)13
Choisir une lettre identifiant le motif voulu?
```

```
F : Fenetre
L : Losange
C : Coin
X : Le X
```

```
Attention, la lettre choisie doit-etre soit F, L, C ou X
```

```
Voulez-vous encore faire un choix de motif O/N?
```

```
Saisir la dimension de la liste (entre 5 et 39)7
Choix de la dimension de la liste est invalide, la dimension doit-etre impaire (entre 5 et 39)
Saisir la dimension de la liste (entre 5 et 39)
```

3 Modalités d'évaluation

Notez qu'un programme qui n'est pas fonctionnel (qui ne s'exécute pas ou qui plante à l'exécution) pourrait recevoir une note de 0.

Votre projet doit contenir les fichiers nécessaires au nombre de 3 en Python (extension.py) et le fichier correction.txt fourni avec l'énoncé, que vous devez compléter en y indiquant votre nom, votre NI, et le nombre d'heures que vous avez consacré à compléter votre TP.

Vous devez compresser votre projet dans une archive Zip (fichier avec extension .zip). Assurez-vous que vous remettez le bon fichier. Nous ne pourrons pas donner de points à votre travail si vous remettez le mauvais fichier (la procédure du dépôt est la responsabilité de l'étudiant).

4 Remarques

Plagiat: Tel que décrit dans le plan de cours, le plagiat est strictement interdit. Ne partagez pas votre code source à quiconque. Une politique stricte de tolérance zéro est appliquée en tout temps et sous toute circonstance. Tous les cas détectés seront référés à la direction de la faculté. Des logiciels comparant chaque paire de TPs pourraient être utilisés pour détecter les cas de plagiat.

Retards : Tout travail remis en retard peut être envoyé par courriel à l'enseignant si le portail des cours n'accepte pas la remise. Une pénalité de 25% sera appliquée pour un travail remis le lendemain de la remise. Une note de 0 sera attribuée pour un plus long retard. Remises multiples: Il vous est possible de remettre votre TP plusieurs fois sur le portail des cours. Seulement la dernière version sera considérée pour la correction. La correction des TPs ne commence qu'après la date limite du dépôt du TP.

Respect des normes de programmation: Nous vous demandons de prêter attention au respect des normes de programmation établies pour le langage Python, notamment de nommer vos variables et fonctions en utilisant la convention suivante: `ma_variable`, `fichier_entree`, etc. Utiliser PyCharm s'avère être une très bonne idée ici, car celui-ci nous donne des indications sur la qualité de notre code (en indentation, marge à droite, et souligné).

Bon travail!