

ITI 1520 – Test 4

Disponible: le 8 décembre, 2020, 14:30 pendant les heures de la dernière classe.

Durée: 1 heure et 45 minutes, plus 15 min pour le remettre.

SVP notez que le test n'est pas accepté après cette date, le système BrightSpace va fermer à 16:30.

Notes:

Vous devez faire ce travail **individuellement**.

Pendant le test, c'est permis d'utiliser les notes disponibles en BrightSpace (notes de cours, de labo, devoirs), le manuel de cours, et documentation de python.org et de IDLE.

Si vous utilisez autre chose, il faut ajouter une déclaration dans votre soumission en BrightSpace et donner la référence dans un fichier `declaration_VOTRE_NOM.txt`. Voir ci-dessous pour plus des détails.

Le nom du fichier et les noms des fonctions doivent être les noms requis, parce qu'on va utiliser des tests automatiques pour la correction. Si le nom d'une fonction n'est pas le nom requis, la note sera zéro pour la fonction. Si le fichier .py donne erreur de syntaxe, la note sera zéro pour les questions dans le fichier. Si vous envoyez le test plusieurs fois, la dernière version sera notée.

SVP noter qu'on va utiliser un outil logiciel pour détecter du plagiat. En cas deux tests sont identique ou très similaires, la note sera zéro pour les deux. Pendant le test ITI1520 les serveurs Discord, Chegg, et StackOverflow sur Internet seront surveillés par les assistants.

Barème : total de 40 points. Test 4 est 15% de la note finale.

Test 4:

Télécharger le fichier `test4_XXXXXX.py`

Remplacez XXXXXX dans le nom de fichier avec `VOTRE_NUMERO_ETUDIANT`.

Ouvrez le fichier en IDLE et ajoutez votre code dans les endroits indiqués. Ce n'est pas permis de modifier le code donné.

Vous devez remettre en BrightSpace un seul fichier, le fichier

`test4_VOTRE_NUMERO_ETUDUANT.py` avec le code requis. Il y a une question plus longue avec des classes et objets à compléter et deux questions courtes avec des fonctions récursives à compléter.

Testez avec votre programme principal ou dans l'interpréteur Python pour vous-même. **Pas besoin de soumettre des tests.**

Ce n'est pas permis d'utiliser des variables globales dans vos fonctions ou méthodes.

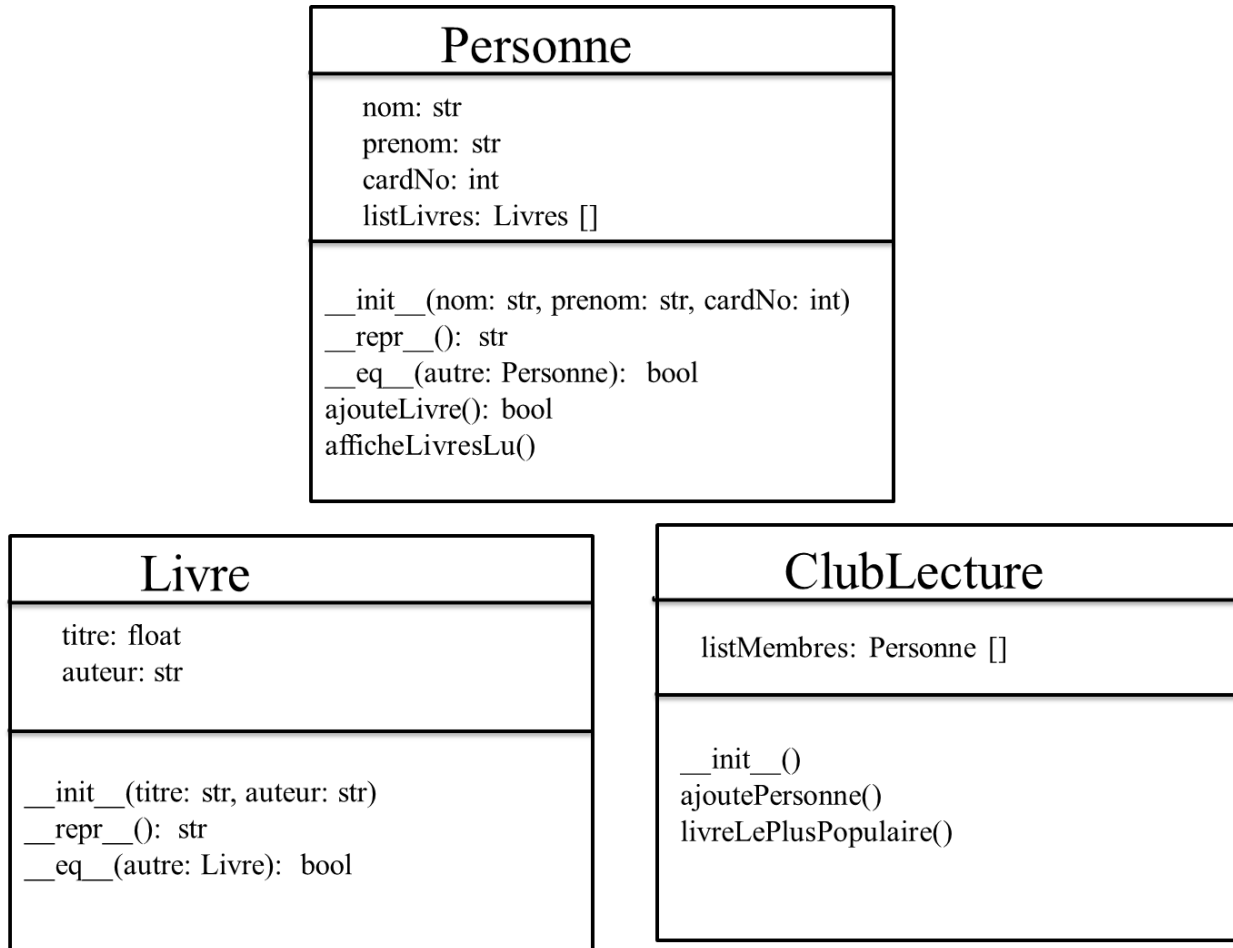
Modules incluses 1-11 avec emphase sur modules 10 et 11.

Question 1 (20 points)

Considérez les classes suivantes, décrites dans le diagramme UML. Complétez-les dans le fichier `test4_XXXXXX.py`. Il y a la description des méthodes dans le fichier, en commentaires. Le programme principal est donné.

Il y a une class `Livre`, et chaque livre a un titre et un auteur. Il a une classe `ClubLecture`, qui a une liste de membres qui sont des objets de classe `Personne`. La class `Personne` contient le nom, prenom, numéro de carte de membre dans le club, et la liste de livre lu par la personne. Le code des constructeurs est donné.

Voir deux exemples d'exécution du programme principal dans les fichiers exemple_run1.txt et exemple_run2.txt



Question 2 (10 points)

Écrivez une fonction Python récursive appelée `sommeLRec` qui prendra comme paramètre une liste et comme deuxième paramètre le nombre des éléments de la liste, et qui retournera la somme des éléments avec de valeurs paires. *Si la fonction n'est pas récursive, le nombre des points pour cette question sera zero.*

Exemples :

```
>>> l=[1,2,5,6,4]
>>> sommeLRec(l, len(l))
12
>>> l=[1,3]
>>> sommeLRec(l, len(l))
0
>>> l=[]
>>> sommeLRec(l, len(l))
0
>>> l=[2,4,8]
```

```
>>> sommeLRec(1, len(1))
14
```

Question 3 (10 points)

Écrivez une fonction `draw_stars_rec(n)` qui dessine les images suivantes, où n est le nombre des étoiles dans le haut et dans le bas de l'image. Par exemple, si n est 9, on dessine 9 étoiles, après ça 7, 5, 3, 1, 3, 5, 7, et 9 étoiles. On fait l'hypothèse que n est un entier positif impair.

Si la fonction n'est pas récursive, le nombre des points pour cette question sera zéro. Vous pouvez utiliser une boucle pour dessiner une ligne d'étoiles, mais pas le dessin entier. Ou vous pouvez utiliser la multiplication des chaînes des caractères pour dessiner une ligne d'étoiles.

Exemples:

```
>>> draw_stars_rec(9)
*****
*****
*****
***
*
***
*****
*****
*****
>>> draw_stars_rec(7)
*****
*****
***
*
***
*****
*****
>>> draw_stars_rec(5)
*****
***
*
***
*****
>>> draw_stars_rec(3)
***
*
***
>>> draw_stars_rec(1)
*
```

Notes pour le fichier `declaration_VOTRE_NOM.txt` :

Ce fichier doit contenir la référence pour le code qui n'est pas écrit par vous-même si c'est le cas. Ça inclut le code donné par un collègue ou autre personne, ou trouvé sur l'internet, réseaux sociaux (comme Stack Overflow, discord, chegg ou autre). Ça n'inclut pas le code donné en BrightSpace dans les notes de classe, labo, etc.

Pour chaque question/fonction où vous avez utilisé du code donné ou trouvé, il faut:

1. le numéro de question
2. copier-coller le code emprunter. Ça inclue code donné/trouvé et modifier très peu.
3. le nom de la source: personne, site Internet ou autre

Vous allez perdre des points la question, mais au moins vous évitez une accusation de plagiat. En cas de plagiat, la note est zéro et le cas doit être rapporté au doyen. Le même est valable si quelqu'un montre son code a un collègue.

Si vous n'utilisez pas cette déclaration, c'est équivalent à déclarer que tous le code a été écrit par vous-même.