

ITI 1520 - Devoir 3

Disponible: le 10 oct, 2024

Date de remise: mardi, le 29 oct, 2024, 22:00

SVP notez que le devoir n'est pas accepté après cette date.

Vous devez faire ce travail **individuellement**. Vous devez soumettre un répertoire compressé en BrightSpace (le fichier d3_xxxx.zip ou xxxx est votre numéro d'étudiant) contenant un fichier d3.py pour Part1 et le fichier d3Part2.py pour Part2 (ce fichier est donné mais vous devez le compléter).

Les noms des fichiers 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. Ajouter des commentaires pour chaque fonction (avec une description courte et contrat de type). Si votre fichier d3.py donne erreur de syntaxe, la note sera zéro.

Si vous envoyez le devoir plusieurs fois, la dernière version sera notée.

SVP noter qu'on va utiliser un outil logiciel pour détecter du plagiat. En cas deux devoirs sont identiques ou très similaires, la note sera zéro pour les deux.

Le but de ce devoir est de tester les concepts de Module 5 et Module 6.

Ajouter une déclaration dans un fichier **declaration_VOTRE_NOM.txt**. avec votre soumission en BrightSpace (dans le fichier d3_xxxxx.zip)

Notes pour le fichier **declaration_VOTRE_NOM.txt** :

Ce fichier doit contenir la référence pour code qui n'est pas écrit par vous même **si c'est le cas**.

Ça inclue code donné par un collègue ou autre personne, ou trouvé sur l'internet, réseaux sociale (comme **Chegg, CourseHero, Stack Overflow, discord**, ou autre), ou généré par **ChatGPT** ou autre assistant virtuel. Ça n'inclue pas le code donné en BrightSpace dans les notes de classe, labo, etc.

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

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

Vous pouvez perdre des points pour 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 à un collègue.

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

Barème : total de **24** points. Devoir 3 est 4% de la note finale.

Part1.

Question 1. (3 points) Créez une fonction Python appelée `compte100` qui prend une liste de nombres et retourne le nombre d'éléments supérieurs à 100 trouvés dans la liste. Ajoutez en commentaires le contrat de type (devinez-le vous-même). Voir des exemples des tests ci-dessous.

Question 2. (3 points) Créez une fonction en Python nommée `sommeListeDiv2` qui prend comme entrée une liste des entiers et retourne la somme des éléments de la liste qui se divisent par 2. N'oubliez pas d'ajouter en commentaires le contrat de type.

Question 3. (3 points) Créez une fonction Python appelée `triples` qui prend une chaîne de caractères et qui retourne `True` s'il y a au moins une séquence de 3 caractères consécutifs égaux, et `False` dans le cas contraire. Assurez-vous que la fonction est efficace ; elle doit s'arrêter dès que le résultat est connu (des qu'on a trouvé 3 caractères consécutifs avec la même valeur). N'oubliez pas d'ajouter en commentaires le contrat de type.

Question 4. (3 points) Créez une fonction Python appelée `momo` qui prend une chaîne de caractères et qui retourne une nouvelle chaîne de caractères qui contient les caractères de la chaîne donnée une fois chacun (sans répétitions des caractères consécutifs), en même ordre, et avec leur nombre de répétitions. Par exemple, si la chaîne donnée est `'bbcggggeeecbb'`, le résultat est `'b2c1g4e5b2'`. N'oubliez pas d'ajouter en commentaires le contrat de type.

Question 5. (3 points) Créez une fonction Python appelée `noDup` qui prend une chaîne de caractères et qui retourne une nouvelle chaîne de caractères qui contient les caractères de la chaîne donnée une fois chacun (sans répétitions des caractères consécutifs), en même ordre, sans répétitions. Par exemple, si la chaîne donnée est `'bbcggggeeec'`, le résultat est `'bcge'`. N'oubliez pas d'ajouter en commentaires le contrat de type.

Testez les fonctions dans l'interpréteur Python ou avec le unit test fourni.

Voici quelques cas de test (testez aussi avec autres valeurs).

Optionnel : Ajouter un programme principal dans votre fichier `d3.py` pour appeler vos fonctions avec les valeurs ci-dessous.

```
#test Q1
compte100([1,102,-3.5, 104])
2
compte100([1,99,-3.5,-7])
0
compte100([])
0

#test Q2
sommeListeDiv2([1,4,3,8,5])
12
sommeListeDiv2([])
0
sommeListeDiv2([4,-10,7])
-6

#test Q3
triples("abc")
```

```

False
triples("abbbcddeegggg")
True
triples("abc2eee")
True
triples("a23xxxxx")
True
triples("abaacd")
False

#test Q4
momo("a")
'a1'
momo("aabbcbccccx")
'a2b3c4x1'
momo("aaa1111")
'a314'
momo("aaabcaax")
'a3b1c1a2x1'

#test Q5
noDup("a")
'a'
noDup("aabbcbccccx")
'abcx'
noDup("aaa1111")
'a1'
noDup("aaabcaax")
'abcax'

```

Part 2 (9 points) Jeu

Cette question est un jeu de cartes, appelé « Pouilleux ». Voir la description complète a [https://fr.wikipedia.org/wiki/Pouilleux_\(jeu\)](https://fr.wikipedia.org/wiki/Pouilleux_(jeu))
(version en anglais : <http://www.classicgamesandpuzzles.com/Old-Maid.html>)

Description du jeu (simplifiée): Le pouilleux est un jeu qui nécessite 51 cartes, c'est-à-dire les 52 cartes traditionnelles moins la carte du valet de trèfle. Chaque joueur prend une carte de son voisin de gauche, sans voir la carte. Ensuite, toutes les paires sont défaussées. Une paire c'est deux cartes avec la même valeur. Le joueur qui se retrouve avec le pouilleux, c'est-à-dire un valet, a perdu.

Pour simplifier le jeu encore plus, on aura seulement deux joueurs: l'ordinateur (i.e., votre programme) et un usager (vous). On va appeler l'ordinateur Robot et l'usager Humain. Robot va donner les cartes pour chaque jeu.

Une partie du jeu est déjà fournie dans le fichier d3Part2.py. Vous pouvez l'étudier et le tester. Ensuite, vous devez ajouter votre code dans le même fichier, aux endroits indiqués par des commentaires, dans les fonctions qui ont seulement des commentaires « docstring » et le code partial ou pas de code.

Après le compléter, ajouter votre fichier d3Part2.py dans le fichier zip pour le soumettre avec Part 1 de Devoir 3.

Il n'est pas permis de modifier les fonctions qui sont déjà complétées, ni le programme principal. On ne peut changer que les fonctions qui ne sont pas complétées. Vous pouvez ajouter d'autres fonctions si vous le souhaitez.

Les fonctions `prepare_paquet`, `attend_le_joueur` et `melange_paquet` sont déjà complétées. Vous devez compléter les autres fonctions: `donne_cartes`, `elimine_paires`, `affiche_cartes`, `entrez_position_valide`, et `joue`. Ces fonctions doivent respecter les descriptions pour le type de données et le type de résultat (les commentaires « docstring »). Ne pas modifier leur en-tête (nom et nombre des paramètres).

La plupart de votre code doit aller dans la fonction `joue`, qui devrait appeler les autres fonctions (i.e., `donne_cartes`, `elimine_paires`, `affiche_cartes`, et `entrez_position_valide`).

Pour la fonction `entrez_position_valide`, on énonce l'hypothèse que Humain va entrer une valeur entière, mais vérifier dans la fonction si c'est un entier valide (entre 1 et le nombre de cartes). La fonction devrait demander à Humain quelle carte il veut, parmi les cartes de Robot que Humain ne peut pas voir. Quand Robot prend une carte de Humain, la carte doit être choisie de manière aléatoire (vous pouvez utiliser la fonction `random`). Si Human demande, par exemple la 3ème carte, cela signifie l'élément de position/index 2 parmi les cartes de Robot (parce que c'est une liste).

Suggestions:

- Étudier les parties du code déjà complétées pour les comprendre.
- Penser aux parties du jeu qui restent à compléter. Par exemple, le jeu doit montrer à Humain ces cartes, être capable de demander à Humain quelle carte il veut, enlever les paires de carte pour Robot et pour Humain, etc.
- Compléter les fonctions une à la fois, pour éviter trop d'erreurs en même temps. Vous pouvez tester chaque fonction avant de compléter les autres, dans l'interpréteur Python. Par exemple, vous pouvez tester la fonction `elimine_paires` avec des cas de test comme celui-ci: (l'ordre des éléments dans le résultat pourrait être différent):

```
>>> elimine_paires(['10♣', '2♣', '5♦', '6♣', '9♣', 'A♦', '10♦']):  
['2♣', '5♦', '6♣', '9♣', 'A♦']
```

- Le jeu prend des tours entre Robot et Humain. Pensez à la façon dont vous pouvez représenter chaque tour (à qui de jouer). Une possibilité consiste à utiliser une variable qui est zéro quand c'est le tour de Robot et 1 quand c'est celui de Humain. Vous devrez aussi déceler à quel moment le jeu prend fin et qui a gagné.

Exemples d'exécutions:

```
=====
Jeu 1
=====
Bonjour. Je m'appelle Robot et je distribue les cartes.
Votre main de cartes est:
7♣ 9♦ 10♣ 9♠ J♠ 8♣ 3♥ Q♦ 6♥ J♦ 3♦ 6♠ Q♥ 7♦ 7♥ 5♣ Q♠ 9♥ K♠ 6♣ 5♦ 2♠ A♣ 4♥ 4♣
2♣
Ne vous inquiétez pas, je ne peux pas voir vos cartes ni leur ordre.
Maintenant défaussez toutes les paires de votre main. Je vais le faire moi
aussi.
Appuyez Enter pour continuer.
*****
Votre tour.
Votre main est:
10♣ 9♦ 6♣ 7♣ Q♦ 8♣ K♠ A♣
J'ai 9 cartes. Si 1 est la position de ma première carte et
9 la position de ma dernière carte, laquelle de mes cartes voulez-vous?
```

SVP entrez un entier de 1 à 9: 1
 Vous avez demandé ma 1ère carte.
 La voilà. C'est un A♦
 Avec A♦ ajouté, votre main est:
 10♣ 9♦ 6♣ 7♣ Q♦ 8♣ K♠ A♣ A♦
 Après avoir défaussé toutes les paires et mélangé les cartes, votre main est:
 6♣ 7♣ K♠ Q♦ 8♣ 10♣ 9♦
 Appuyez Enter pour continuer.

 Mon tour.
 J'ai pris votre 6ème carte.
 Appuyez Enter pour continuer.

 Votre tour.
 Votre main est:
 6♣ 7♣ K♠ Q♦ 8♣ 9♦
 J'ai 7 cartes. Si 1 est la position de ma première carte et
 7 la position de ma dernière carte, laquelle de mes cartes voulez-vous?
 SVP entrez un entier de 1 à 7: 6
 Vous avez demandé ma 6ème carte.
 La voilà. C'est un 7♠
 Avec 7♠ ajouté, votre main est:
 6♣ 7♣ K♠ Q♦ 8♣ 9♦ 7♠
 Après avoir défaussé toutes les paires et mélangé les cartes, votre main est:
 6♣ K♠ Q♦ 8♣ 9♦
 Appuyez Enter pour continuer.

 Mon tour.
 J'ai pris votre 4ème carte.
 Appuyez Enter pour continuer.

 Votre tour.
 Votre main est:
 6♣ K♠ Q♦ 9♦
 J'ai 5 cartes. Si 1 est la position de ma première carte et
 5 la position de ma dernière carte, laquelle de mes cartes voulez-vous?
 SVP entrer un entier de 1 à 5: 5
 Vous avez demandé ma 5ème carte.
 La voilà. C'est un 6♦
 Avec 6♦ ajouté, votre main est:
 6♣ K♠ Q♦ 9♦ 6♦
 Après avoir défaussé toutes les paires et mélangé les cartes, votre main est:
 9♦ Q♦ K♠
 Appuyez Enter pour continuer.

 Mon tour.
 J'ai pris votre 1ère carte.
 Appuyez Enter pour continuer.

 Votre tour.
 Votre main est:
 Q♦ K♠

J'ai 3 cartes. Si 1 est la position de ma première carte et
3 la position de ma dernière carte, laquelle de mes cartes voulez-vous?
SVP entrer un entier de 1 à 3: 2
Vous avez demandé ma 2ème carte.
La voilà. C'est un Q♣
Avec Q♣ ajouté, votre main est:
Q♦ K♠ Q♣
Après avoir défaussé toutes les paires et mélangé les cartes, votre main est:
K♠
Appuyez Enter pour continuer.

Mon tour.
J'ai pris votre 1ère carte.
Appuyez Enter pour continuer.

J'ai terminé toutes les cartes.
Félicitations! Vous, Humain, vous avez gagné.

=====

Jeu 2

=====

Bonjour. Je m'appelle Robot et je distribue les cartes.
Votre main de cartes est:
6♠ Q♥ Q♣ 9♥ A♥ 3♠ 4♣ 6♣ J♦ 10♦ 5♦ 3♣ J♥ K♠ K♥ A♣ 8♥ 10♣ 4♦ 5♣ 3♥ 10♠ 3♦ 7♦
Q♠ J♠
Ne vous inquiétez pas, je ne peux pas voir vos cartes ni leur ordre.
Maintenant défaussez toutes les paires de votre main. Je vais le faire moi
aussi.
Appuyez Enter pour continuer.

Votre tour.
Votre main est:
Q♣ 10♣ J♦ 7♦ 9♥ 8♥
J'ai 5 cartes. Si 1 est la position de ma première carte et
5 la position de ma dernière carte, laquelle de mes cartes voulez-vous?
SVP entrez un entier de 1 à 5: 7
Position invalide. SVP entrez un entier de 1 à 5: 5
Vous avez demandé ma 5-ème carte.
La voilà. C'est un 8♣
Avec 8♣ ajouté, votre main est:
Q♣ 10♣ J♦ 7♦ 9♥ 8♥ 8♣
Après avoir défaussé toutes les paires et mélangé les cartes, votre main est:
10♣ 7♦ J♦ Q♣ 9♥
Appuyez Enter pour continuer.

Mon tour.
J'ai pris votre 2ème carte.
Appuyez Enter pour continuer.

Votre tour.
Votre main est:
10♣ J♦ Q♣ 9♥

J'ai 3 cartes. Si 1 est la position de ma première carte et
3 la position de ma dernière carte, laquelle de mes cartes voulez-vous?
SVP entrer un entier de 1 à 3: 2

Vous avez demandé ma 2ème carte.

La voilà. C'est un 9♣

Avec 9♣ ajouté, votre main est:

10♣ J♦ Q♣ 9♥ 9♣

Après avoir défaussé toutes les paires et mélangé les cartes, votre main est:

J♦ Q♣ 10♣

Appuyez Enter pour continuer.

Mon tour.

J'ai pris votre 3ème carte.

Appuyez Enter pour continuer.

Votre tour.

Votre main est:

J♦ Q♣

J'ai 1 carte. Si 1 est la position de ma première carte et

1 la position de ma dernière carte, laquelle de mes cartes voulez-vous?

SVP entrer un entier de 1 à 1: 1

Vous avez demandé ma 1ère carte.

La voilà. C'est un Q♦

Avec Q♦ ajouté, votre main est:

J♦ Q♣ Q♦

Après avoir défaussé toutes les paires et mélangé les cartes, votre main est:

J♦

Appuyez Enter pour continuer.

J'ai terminé toutes les cartes.

Vous avez perdu! Moi, Robot, j'ai gagné.