

## ITI 1520 - Devoir 1

Disponible: le 18 septembre, 2020

Date de remise: lundi, le 28 septembre, 2020, 8:00

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 d1.zip) contenant les fichiers d1.py et d1.txt

Mettez toutes les fonctions requises ci-dessous dans le fichier d1.py et les résultats de vos tests dans le fichier d1.txt.

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 d1.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 de deux devoirs identiques ou très similaires, la note sera zéro pour les deux.

Le but de ce devoir est de tester les concepts de Module 1 et Module 2. Vous n'avez pas besoin d'utiliser des boucles et les instructions de branchement (`if`).

Barème : total de 18 points. Devoir 1 est 4% de la note finale.

**Question 1.** (2 points) Écrivez une fonction Python appelée `aireDuTriangle`, qui prend deux variables (la base et la hauteur) et retourne l'aire (la surface) du triangle. Le contrat de type de la fonction est : `(float,float)->float`. Ça signifie que les deux variables données comme entrée sont des nombres réels, et le résultat est un nombre réel.

**Question 2.** (2 points) Écrivez une fonction Python appelée `aireDuTrianglePrint`, qui lit deux variables (la base et la hauteur) du clavier et affiche un message avec l'aire (la surface) du triangle. Le contrat de type de la fonction est : `()->None`. Ça signifie qu'il n'y a pas de valeurs données comme entrée et pas de résultat retourné. La fonction doit lire le clavier avec la fonction Python `input` et ensuite utiliser la fonction de la question 1 pour le calcul de l'aire.

**Question 3.** (2 points) Écrivez une fonction Python appelée `loEnKilos` pour convertir en kilogrammes le poids donné en livres et onces. La fonction prend deux variables comme entrée et retourne comme résultat le nombre de kilogrammes.

Le contrat de type de la fonction est: `(float, float) -> float`

Note: le résultat peut avoir n'importe quel nombre des décimales. C'est important que les 3 premières décimales soit correctes.

**Question 4.** (2 points) Écrivez une fonction Python avec 5 valeur d'entrée `bibformat(auteur,titre,ville,maisonEdition,annee)` qui retourne une chaîne de caractères en format: `auteur (annee). titre. ville: maisonEdition` (voir des exemples dans la section des tests ci-dessous). Le contrat de type de la fonction est: `(str,str,str,str,str)-> str`.

**Question 5.** (2 points) Écrivez une fonction Python `bibformatPrint()` qui lit du clavier le titre d'un livre, le nom de l'auteur, l'année de publication, la maison d'édition et la ville de la maison d'édition. Le contrat de type de la fonction est: `()->None`. La fonction doit afficher l'information sur le livre dans le format spécifié dans la question 4. Vous devez utiliser la fonction `bibformat` de la question 4 pour obtenir la chaîne de caractères à imprimer.

**Question 6.** (2 points) Écrivez une fonction Python appelée `funct` qui prend comme entrée un entier `p` supérieur ou égal à 10 et trouve la solution `r` de l'équation suivante :  $r^2 - p + 10 = 0$ . La fonction affiche le message "La solution est `r`", où `r` est remplacé par la valeur de `r` qui est la solution de l'équation. A la fin, la fonction retourne la valeur de `r`. Le contrat de type de la fonction est: `(int)->float`

**Question 7.** (2 points) Écrivez une fonction Python appelée `magique` qui prend comme entrée un entier positif `n` de trois chiffres. La fonction retourne `True` si `n` est un nombre magique, et `False` sinon. Un nombre est magique si deux chiffres consécutifs sont égaux à 3, ou si le dernier chiffre se divise par 4. Pour cette fonction, ce n'est pas permis d'utiliser des chaînes de caractères. La fonction devrait retourner une expression booléenne. Utilisez modulo et division entière pour déterminer les chiffres. Le contrat de type de la fonction est: `(int) -> boolean`

**Question 8.** (4 points) Combien de pièces?

Supposons qu'un caissier doit rendre la monnaie à un client et qu'il a seulement de la petite monnaie: des pièces de 25 cents, 10 cents, 5 cents, et 1 cent (en anglais: quarters, dimes, nickels, pennies). Écrivez une fonction Python appelée `numMonnaies` pour calculer le nombre minimal de pièces que le caissier peut donner.

La fonction prend comme entrée la valeur en dollars. Utilisez l'algorithme suivant: multipliez la valeur en dollars avec 100 pour la transformer en centimes. Ensuite, calculez le nombre possible de pièces de 25 cents. S'il y a un reste, calculez le nombre de pièces de 10 cents, puis de 5 cents, et enfin de 1 cent, au besoin. La fonction doit afficher le nombre des pièces de 25 cents, 10 cents, 5 cents, et 1 cent, et à la fin retourner le nombre total des pièces.

Utilisez modulo et division entière pour les calculs.

Le contrat de type de la fonction est: `(float) -> int`

Exemples: Si le caissier doit rendre 0,56 dollar, le nombre minimal de pièces est 4 (2 pièces de 25 cents, 0 pièce de 10 cents, 1 pièce de 5 cents and 1 pièce de 1 cent). Il n'est pas possible de donner 3 pièces (ou moins). Si le montant à rendre est 1,42 dollar, le nombre minimal de pièces est 9 (5 pièces de 25 cents, 1 pièce de 10 cents, 1 pièce de 5 cents et 2 pièces de 1 cent).

**Testez les fonctions** dans l'interpréteur Python et ajoutez les résultats dans le fichier `d1.txt`

Copiez et collez dans le fichier `d1.txt` quand vous testez chaque fonction dans l'interpréteur. Votre fichier `d1.txt` devrait contenir quelque chose comme le suivant:

```
>>>#test Q1
>>> aireDuTriangle(20,20)
200.0
>>> aireDuTriangle(40.5, 30.3)
613.575
```

```

>>>
>>>#test Q2
>>> aireDuTrianglePrint()
SVP entrez la valeur de la base: 20
SVP entrez la valeur de la hauteur: 20
Un triangle de base 20.0 et hauteur 20.0 a l'aire 200.0
>>> aireDuTrianglePrint()
SVP entrez la valeur de la base: 40.5
SVP entrez la valeur de la hauteur: 30.3
Un triangle de base 40.5 et hauteur 30.3 a l'aire 613.575
>>>
>>>#test Q3
>>> loEnKilos(5,3)
2.35305
>>> loEnKilos(6.8,3.3)
3.178035
>>>
>>>#test Q4
>>> bibformat("Antoine de Saint Exupery", "Le petit prince", "Paris", "Jeunesse", 1943)
' Antoine de Saint Exupery (1943). Le petit prince. Paris: Jeunesse'
>>>
>>>#test Q5
>>> bibformatPrint()
SVP entez l'auteur: Antoine de Saint Exupery
SVP entez le titre: Le petit prince
SVP entez la ville: Paris
SVP entez la maison d'edition: Jeunesse
SVP entez l'annee de publication: 1943
Antoine de Saint Exupery (1943). Le petit prince. Paris: Jeunesse
>>>
>>>#test Q6
>>> funct(10)
La solution est 0.0
0.0
>>> funct(49)
La solution est 6.244997998398398
6.244997998398398
>>> funct(59)
La solution est 7.0
7.0
>>>
>>>#test Q7
>>> magique(303)
False
>>> magique(330)
True
>>> magique(133)
True
>>> magique(712)
False
>>> magique(128)
True
>>>
>>>#test Q8
>>> numMonnaies(1.56)
6 pieces de 25 cents, 0 pieces de 10 cents, 1 pieces de 5 cents, 1 pieces de 1 cent
8
>>> numMonnaies(1.42)
5 pieces de 25 cents, 1 pieces de 10 cents, 1 pieces de 5 cents, 2 pieces de 1 cent
9
>>> numMonnaies(1)
4 pieces de 25 cents, 0 pieces de 10 cents, 0 pieces de 5 cents, 0 pieces de 1 cent
4

```