

Année Universitaire:	2022-2023	Date de l'Examen:	28/11/2022
Nature:	DC	Durée:	1h30min
Diplôme:	Ingénieur	Nombre de pages:	2
Section:	GEA	Enseignant:	LASSOUED Zeineb
Niveau d'études:	2ème année	Doc autorisés:	Non
Matière:	Programmation avancée	Remarque:	

## Exercice 1 (10 points)

Avec Python, répondre aux questions suivantes :

1. Saisir successivement 7 nombres entiers et afficher la liste des nombres saisis tout en ignorant les nombres répétés.

Exemple, si l'utilisateur saisi les nombres : 2 , 11 , 3 , 2 , 3 , 5 , 2. l'algorithme renvoie la liste **Entiers**= [2 , 11 , 3 , 5].

2. Créer une liste **Nombres** dont les éléments varient entre 0 à 9 en une seule ligne.
3. Ajouter la liste **Nombres** à la fin de la liste **Entiers**.
4. Inverser les éléments de la liste **Entiers**.
5. Utiliser une liste en compréhension pour multiplier par 2 chaque élément de la liste **Entiers** mais seulement si l'élément est impair.
6. Créer un fichier nommé "myFile.txt" et y écrire le texte suivant : "Python est le meilleur langage de programmation" puis supprimer le 4ème mot du texte ajouté.
7. Lire un fichier déjà créé nommé "myFile.txt" et construire un dictionnaire dont les clés sont les mots qui composent le texte et les valeurs sont les longueurs des mots.
8. On considère un dictionnaire dont les clés sont les noms des élèves et les valeurs des clés sont les moyennes générales obtenues en passant l'examen final. Écrire un programme qui partitionne ce dictionnaire en deux sous dictionnaires :  
**etudiantAdmis** dont les clés sont les étudiants admis et les valeurs des clés sont les moyennes obtenues (moyenne supérieurs ou égales à 10 ).  
**etudiantNonAdmis** dont les clés sont les étudiants non admis et les valeurs des clés sont les moyennes obtenues (moyenne inférieur ou égale à 10).



## Exercice 2 (10 points)

Avec Python, programmer les fonctions suivantes :

1. Écrire une fonction qui prend en paramètre une liste  $L$  et qui renvoie la liste des éléments dupliqués au moins 3 fois sans utiliser la méthode *count* ni aucune méthode prédéfinie en Python.
2. Étant donnée un dictionnaire  $d$  dont les valeurs des clés sont des listes de nombres. Écrire une fonction qui permet de transformer le dictionnaire  $d$  en triant les listes.  
Exemple pour le dictionnaire  $d$  :  
 $d = \text{'a1'} : [21, 17, 22, 3], \text{'a2'} : [11, 15, 8, 13], \text{'a3'} : [7, 13, 2, 11], \text{'a4'} : [22, 14, 7, 9]$   
La fonction doit renvoyer le dictionnaire :  
 $d = \text{'a1'} : [3, 17, 21, 22], \text{'a2'} : [8, 11, 13, 15], \text{'a3'} : [2, 7, 11, 13], \text{'a4'} : [7, 9, 14, 22]$
3. Écrire une fonction qui prend en paramètre une chaîne de texte  $T$  et qui renvoie la liste des mots contenant au moins un chiffre et une majuscule.
4. Écrire en utilisant la méthode *count()*, une fonction qui prend en paramètre une liste d'entiers  $L$  et qui renvoie sans répétitions la liste des tuples  $(n, \text{occ-n})$  formée des éléments  $n$  de  $L$  et de leurs occurrence *occ-n*.  
Exemple : si  $L = [22, 7, 14, 22, 7, 14, 7, 14, 11, 7]$ , l'algorithme renvoie la liste  $[(22, 2), (7, 4), (14, 3), (11, 1)]$
5. Écrire une fonction qui prend en arguments deux nombres entiers  $a$  et  $b$  et qui renvoie un couple d'entiers formé par :
  - \* Le quotient  $q$  de la division euclidienne de  $a$  par  $b$  (sans utiliser l'opérateur `//`)
  - \* Le reste  $r$  de la division euclidienne de  $a$  par  $b$  (sans utiliser l'opérateur `%`)



Année Universitaire:	2022-2023	Date de l'Examen:	09/01/2023
Nature:	Examen	Durée:	2h
Diplôme:	Ingénieur	Nombre de pages:	2
Section:	GEA	Enseignant:	LASSOUED Zeineb
Niveau d'études:	2ème année	Doc autorisés:	Non
Matière:	Programmation avancée : Python	Remarque:	

## Exercice 1 (8 points)

Avec le langage de programmation Python, répondre aux questions indépendantes suivantes :

1. Étant donné le dictionnaire Python suivant qui contient les noms des élèves avec leurs notes à l'examen :

Results = {'Maria' : 9, 'Amelia' : 12, 'Hafid' : 16, 'Hajar' : 11, 'Nathan' : 14}

Écrire un programme qui permet d'améliorer les notes en ajoutant 2 points à chaque élève.

2. Demander à l'utilisateur de saisir une liste d'entiers sachant qu'en cas d'erreur l'exécution ne doit pas être planter. En plus, la liste doit être imprimée à l'écran à chaque ajout réussi d'un élément.
3. Écrire un algorithme qui détermine pour un entier  $n$  donné la liste des Tuples d'entiers  $(p,q)$  vérifiant :  $p^2 + q^2 = n$
4. Tracer le graphe de la fonction  $y(t) = 10t - \frac{1}{2}gt^2$  pour  $g = 9.81$ . Le label sur l'axe des abscisses devra être "temps (s)" et le label sur l'axe des ordonnées est "hauteur (m)".

## Exercice 2 (6 points)

Avec Python, développer les fonctions suivantes :

1. Une fonction qui prends en paramètre une liste de nombres entiers L et qui renvoie la liste obtenue à partir de L en insérant juste après chaque nombre la chaîne de caractères 'pair' ou 'impair' selon la parité de nombre.

Exemple si

$L = [2, 11, 25, 6, 14]$ , l'algorithme renvoie la liste :

$[2, 'pair', 11, 'impair', 25, 'impair', 6, 'pair', 14, 'pair']$



2. Une fonction qui renvoie la liste des diviseurs d'un entier donné. Exemple si  $n = 18$ , l'algorithme renvoie la liste  $[1, 2, 3, 6, 9, 18]$
3. Une fonction qui prends en paramètre une liste de nombres entiers  $L$  et qui renvoie la liste obtenue à partir de  $L$  en déplaçant tous les zéros au début de la liste.

### Exercice 3 (6 points)

On désire réaliser en python une classe nommée **Cercle**.

1. Définir une classe **Cercle** permettant de créer un cercle  $C(O,r)$  de centre  $O(a,b)$  et de rayon  $r$  à l'aide du constructeur.
2. Définir une méthode **Surface()** de la classe qui permet de calculer la surface du cercle.
3. Définir une méthode **Perimetre()** de la classe qui permet de calculer le périmètre du cercle.
4. Définir une méthode **testAppartenance()** de la classe qui permet de tester si un point  $A(x,y)$  appartient ou non au cercle  $C(O,r)$ .
5. Enrichir la classe en lui ajoutant une méthode d'affichage en utilisant la méthode spéciale **str**.
6. Développer un programme principale permettant d'instancier des objets de type **Cercle** et tester les fonctions déjà établies.