

EXAMEN DE FIN DU SEMESTRE 1

Matière : INFORMATIQUE

Classes : 1^{ère} Année MP – T – PC Durée : 2 h

N.B. :

- Toutes les réponses doivent être rédigées en Python.
- L'indentation dans les instructions Python sera prise en considération dans le barème.
- Le sujet comporte 4 pages.
- Vous trouvez, en *Annexe*, quelques méthodes prédéfinies sur les conteneurs.

Exercice 1 (4 points)

La constante de Catalan **k** est le nombre défini par :

$$K = \sum_{n=0}^{+\infty} \frac{(-1)^n}{(2n+1)^2} = 1 - \frac{1}{3^2} + \frac{1}{5^2} - \frac{1}{7^2} + \dots$$

Le calcul est arrêté lorsque la valeur absolue du dernier terme calculé est inférieure à la précision voulue $\varepsilon = 10^{-6}$.

Écrire un script Python permettant de calculer et d'afficher la valeur approchée de la constante de Catalan **k**.

Exercice 2 (6 points)

Soit la suite U_n définie par :

$$\begin{cases} U_1 = 1 \\ U_2 = 2 \\ U_n = U_{n-1} + U_{n-2} \quad \text{pour tout } n > 2 \end{cases}$$

Écrire un script Python permettant de :

- 1) Saisir un entier **N** étant strictement supérieur à 2.
- 2) Calculer les **N** premiers termes de la suite U_n et les stocker dans une liste **L**.
- 3) Calculer et afficher la somme **S** et la moyenne **M** des **N** premiers termes de la suite U_n .
- 4) Calculer et afficher la variance **V** et l'écart type **E** de la liste **L** précédemment créée, sachant que la variance et l'écart type d'une série de nombres entiers d'une liste **L** (de longueur **N** et de moyenne **M**) sont définies, respectivement, par :

$$V = \frac{1}{N} \sum_{i=0}^{N-1} (L[i] - M)^2 \qquad E = \sqrt{V}$$

Exercice 3 (6 points)

On se propose de crypter un message composé de deux ou plusieurs mots étant séparés par des blancs d'espace, en utilisant le principe suivant (**voir exemple ci-après**) :

- 1- Créer la liste **L** formée par les mots du message à crypter (*étape 1*).
- 2- Modifier la liste **L** en ajoutant, pour chaque mot, autant de fois, le caractère "*" à la fin, pour que sa longueur soit égale à celle du mot le plus long dans cette liste (*étape 2*).
- 3- Créer une nouvelle liste **L1** formée par les caractères du premier mot dans **L**, c'est-à-dire l'élément à l'indice zéro (*étape 3*).
- 4- Modifier la liste **L1** en ajoutant les caractères du deuxième mot se trouvant dans **L** à la liste **L1**, en concaténant à chaque fois le caractère d'indice **i** de **L[1]** avec l'élément **L1[i]** (*étape 4*).
- 5- Procéder de la même façon avec le reste des mots se trouvant dans la liste **L**.
- 6- À la fin, concaténer les chaînes de caractères obtenues dans la liste **L1** en les séparant par le caractère d'espace pour obtenir le message crypté.

Exemple :

Si le message à crypter est "**Partager le savoir ne le divise pas**", alors les étapes du cryptage proposé sont :

Étape 1 : Placer les mots du message à crypter dans la liste **L** ;

L = ["Partager", "le", "savoir", "ne", "le", "divise", "pas "]

Étape 2 : Ajouter le caractère "*" autant de fois pour obtenir des mots dont la longueur de chacun est égale à celle du mot le plus long dans le message.

Étant donné que "**Partager**" est le mot le plus long du message (8 caractères), on obtient alors la liste **L** suivante :

L = ["Partager", "le*****", "savoir**", "ne*****", "le*****", "divise**", "pas*****"]

Étape 3 : Répartir les lettres de **L[0]** dans la liste **L1** ;

L1 = ['P', 'a', 'r', 't', 'a', 'g', 'e', 'r']

Étape 4 : Ajouter les lettres de **L[1]** dans la liste **L1** ;

L1 = ['Pl', 'ae', 'r*', 't*', 'a*', 'g*', 'e*', 'r*']

Après les étapes suivantes :

L1 = ['Plasldp', 'aeaceia', 'r*v**vs', 't*o**i*', 'a*i**s*', 'g*r**e*', 'e*****', 'r*****']

Le message crypté est alors :

"Plasldp aeaceia r*v**vs t*o**i* a*i**s* g*r**e* e***** r*****"

Travail demandé :

Écrire un script Python permettant de :

1. Saisir une chaîne de caractères **msg** composée de deux ou plusieurs mots.
2. Créer la liste **L**, puis la liste **L1**, selon le principe décrit précédemment.
3. Construire et afficher la chaîne de caractère **msgcry** correspondant au message crypté.

Exercice 4 (4 points)

Donner la classe de complexité en O des quatre exemples suivants :

Exemple 1	
En algorithmique	En Python
<pre>lire(n) pour i de 1 à n faire pour j de i-10 à i+10 faire x ← 2 y ← x + 10 finPour finPour</pre>	<pre>n = int(input()) for i in range(1,n+1) : for j in range(i-10,i+11) : x = 2 y = x + 10</pre>

Exemple 2	
En algorithmique	En Python
<pre>lire(n) p ← 0 tantque n > 0 faire n ← n DIV 2 p ← p + 1 fintantque écrire(p)</pre>	<pre>n = int(input()) p = 0 while n > 0 : n = n // 2 p = p + 1 print(p)</pre>

Exemple 3	
En algorithmique	En Python
<pre>lire(n) i ← n tantque i > 1 faire pour j de 1 à n faire x ← 2 * n + 1 y ← (x * x) / 2 finPour i ← i DIV 10 fintantque</pre>	<pre>n = int(input()) i = n while i > 1 : for j in range(1,n+1) : x = 2 * n + 1 y = (x * x) / 2 i = i // 10</pre>

Exemple 4	
En algorithmique	En Python
<pre>lire(n) pour i de 1 à n faire écrire('*') finPour pour i de 1 à (n*n) DIV 2 faire écrire('i=',i) finPour</pre>	<pre>n = int(input()) for i in range(1,n+1) : print('*') for i in range(1, (n*n)//2+1) : print('i=',i)</pre>

Annexe

<u>Méthode</u>	<u>Rôle</u>
L.append(x)	Ajouter l'élément x à la fin de la liste L .
L.pop()	Retourner et supprimer le dernier élément se trouvant dans la liste L .
L.index(x)	Retourner l'indice de la première occurrence de l'élément x dans la liste L s'il existe, sinon, un message d'erreur.
ch.count(c)	Retourner le nombre d'occurrences (ou d'apparition) d'une chaîne c , donnée en argument, dans une chaîne de caractères ch . <u>Exemple</u> : <code>'abccdc'.count('c')</code> retourne 3.
ch.split()	Retourner la liste formée par les sous-chaînes se trouvant dans la chaîne ch et étant séparées par le caractère espace ' '. <u>Exemple</u> : <code>"Hello world !".split()</code> retourne <code>['Hello', 'world', '!']</code>
ch.join(L)	Retourner la chaîne de caractères formée par les chaînes de caractères contenues dans la liste L en les concaténant avec la chaîne de caractères ch <u>Exemple</u> : <code>' '.join(['ipeis', '2023'])</code> retourne la chaîne de caractères <code>'ipeis 2023'</code>