Devoir d'informatique n° TP 1 (1,50 heures)

Ce TP noté est constitué de plusieurs petits exercices. Vous avez le droit d'utiliser les documents de cours ainsi que vos documents personnels mais, bien évidemment, pas ceux de vos voisins. Vous rendrez vos réponses sur la feuille de réponses jointe en y écrivant les valeurs ou réponses aux questions posées. Vous utiliserez Pyzo et effectuerez les imports de toutes les fonctions du module math ainsi que le module numpy avec l'alias np. De plus, pour vérifier le caractère personnel de votre travail, **vous enregistrerez tous les quarts d'heure** vos fichiers Python dans le dossier **Mes devoirs** d'où ils seront ramassés de manière régulière.

Attention : Pour les applications numériques ou d'autres questions, vous possédez chacun un identifiant personnel (M, K) couple de deux nombres donnés dans le tableau suivant :

Eleve	M	K		Eleve	M	K		Eleve	M	K
Baudet	21	13		Chambon	21	14		Ollivier	21	15
Gorisse	21	16		Le Guirriec	21	17		Roger	21	18
Marchand	22	13		Zarrouk	22	14		David	22	15
Dhers	22	16		Ostyn	22	17		Besnard	22	18
Hamon	23	13		Lioux	23	14				

Préliminaires : Utilisation de vos identifiants

Nous allons utilisez vos identifiants (M, K) pour ce TP, afin de déterminer une liste sur laquelle vous effectuerez certains tests des fonctions programmées.

Ecrire des séquences d'instructions permettant d'obtenir la liste **maliste** constituée des 100 entiers démarrant de M et séparés de K . Par exemple si on avait pris M = 10 et K = 2, on aurait eu la liste des 100 entiers pairs compris entre 10 et 208 . Sur la feuille des réponses, vous noterez le premier et le dernier terme de la liste **maliste**.

Exercice 1 : Triplets pythagoriciens

1. Triangles rectangles. On appelle triplet pythagoricien tout triplet (α, β, γ) d'entiers tels que que $0 < \alpha \leqslant \beta \leqslant \gamma$ et $\alpha^2 + \beta^2 = \gamma^2$

   (a) Ecrire une fonction **Paire** prenant comme argument deux entiers a et b et qui renvoie le booléen `True` si (a, b, c) avec $c = \sqrt{a^2 + b^2}$ est un triplet pythagoricien et `False` sinon.

   (b) Ecrire une fonction **Triplets** prenant comme argument un entier naturel n et qui retourne tous les triplets pythagoriciens (a, b, c) avec $0 < a \leqslant b \leqslant n$

   (c) Faites le test avec n valant la somme de vos identifiants M et K. De même, combien y-a-t-il de triplets pythagoriciens (a, b, c) avec $0 < a \leqslant b \leqslant N$ où est le produit de vos identifiants M et K ?

   (d) En utilisant la fonction **Paire** , écrire une fonction **TRPI** prenant comme argument un entier naturel n et qui retourne tous les triplets pythagoriciens de la forme $(a, a+1, c)$ avec $0 < a < n$.

   (e) Tester cette fonction avec $N = M \times K$ où M et K sont vos identifiants.

Exercice 2 : Travail avec les sommes et les produits

1. Somme des inverses des carrés des n premiers entiers $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2}$.

   (a) Ecrire une fonction appelée **zeta** prenant pour argument un entier naturel n et retourne la somme $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2}$.

   (b) Calculer **zeta**(N) où $N = M \times K$ est le produit de vos deux identifiants M et K

   (c) Trouver (avec Python) l'entier n_0 tel que pour tout $n \geqslant n_0$, $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2} > 1.5$

   (d) En déduire $\forall n \in \mathbb{N}^* \setminus \{1\}$, $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2} > \frac{3n}{2n+1}$

2. Paradoxe des anniversaires
   La probabilité p_n qu'au moins deux étudiants d'une classe de $n \geqslant 2$ étudiants aient leur anniversaire le même jour de l'année est donnée par la formule : $p_n = 1 - \displaystyle\prod_{k=1}^{n-1} \left(1 - \frac{k}{365}\right)$

   (a) Ecrire une fonction appelée **anniversaire** prenant pour argument un entier naturel n et qui retourne la valeur p_n

   (b) Tester la fonction pour les entiers M, K et $M + K$ de votre identifiant

   (c) Déterminer (avec Python) l'entier n_0 tel que pour tout $n \geqslant n_0$, $p_n > 0.5$

   (d) Déterminer de même l'entier n_1 tel que pour tout $n \geqslant n_1$, $p_n > 0.9$

3. Sommes doubles

   On rappelle que la somme double $\displaystyle\sum_{1 \leqslant i,j \leqslant 10} i\,j$ se calcule sous Python sous la forme :

```
s = 0
for i in range(1,11):
  for j in range(1,11):
    s = s + i*j
print(s)
```

   (a) Ecrire une fonction appelée **SommeDouble** prenant pour argument un entier naturel n et qui retourne la valeur $S_n = \displaystyle\sum_{1 \leqslant i,j \leqslant n} \max(i\,j)$. (On pourra utiliser la fonction max de Python)

   (b) Tester la fonction pour les entiers M, K et $M + K$ de votre identifiant

   (c) Montrer (avec des tests Python) que pour tous les entiers n de votre liste personnelle **maliste**, on a $S_n = \dfrac{n(n+1)(4n-1)}{6}$. On essaiera de proposer un test qui ne repose pas sur un affichage de toutes les réponses pour les 100 valeurs à tester...

   (d) Ecrire une fonction appelée **SommeTriangulaire** prenant pour argument un entier naturel n et qui retourne la valeur $T_n = \displaystyle\sum_{1 \leqslant i < j \leqslant n} \frac{i}{j}$.

   (e) Tester la fonction pour les entiers M, K et $M + K$ de votre identifiant

   (f) Montrer (avec des tests Python) que pour tous les entiers n de votre liste personnelle **maliste**, on a $T_n = \dfrac{n(n-1)}{4}$

Devoir d'informatique n° TP 1 (1,50 heures)

Ce TP noté est constitué de plusieurs petits exercices. Vous avez le droit d'utiliser les documents de cours ainsi que vos documents personnels mais, bien évidemment, pas ceux de vos voisins. Vous rendrez vos réponses sur la feuille de réponses jointe en y écrivant les valeurs ou réponses aux questions posées. Vous utiliserez Pyzo et effectuerez les imports de toutes les fonctions du module math ainsi que le module numpy avec l'alias np. De plus, pour vérifier le caractère personnel de votre travail, **vous enregistrerez tous les quarts d'heure** vos fichiers Python dans le dossier **Mes devoirs** d'où ils seront ramassés de manière régulière.

Attention : Pour les applications numériques ou d'autres questions, vous possédez chacun un identifiant personnel (M, K) couple de deux nombres donnés dans le tableau suivant :

Eleve	M	K		Eleve	M	K		Eleve	M	K
Louatron	21	13		Leblond	21	14		El Atchia	21	15
Okon	21	16		Plantevin	21	17		Quentin	21	18
Belouah	22	13		Bailly	22	14		Caillebotte	22	15
Caudroit	22	16		Diverd	22	17		Drouet	22	18
Nader	23	13								

Préliminaires : Utilisation de vos identifiants

Nous allons utilisez vos identifiants (M, K) pour ce TP, afin de déterminer une liste sur laquelle vous effectuerez certains tests des fonctions programmées.

Ecrire des séquences d'instructions permettant d'obtenir la liste **maliste** constituée des 100 entiers démarrant de M et séparés de K . Par exemple si on avait pris M = 10 et K = 2, on aurait eu la liste des 100 entiers pairs compris entre 10 et 208 . Sur la feuille des réponses, vous noterez le premier et le dernier terme de la liste **maliste**.

Exercice 1 : Travail avec les sommes et les produits

1. Somme des inverses des carrés des n premiers entiers $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2}$.
   (a) Ecrire une fonction appelée **zeta** prenant pour argument un entier naturel n et retourne la somme $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2}$.
   (b) Calculer **zeta**(N) où $N = M \times K$ est le produit de vos deux identifiants M et K
   (c) Trouver (avec Python) l'entier n_0 tel que pour tout $n \geqslant n_0$, $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2} > 1.5$
   (d) En déduire $\forall n \in \mathbb{N}^* \setminus \{1\}$, $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2} > \frac{3n}{2n+1}$
2. Paradoxe des anniversaires
   La probabilité p_n qu'au moins deux étudiants d'une classe de $n \geqslant 2$ étudiants aient leur anniversaire le même jour de l'année est donnée par la formule : $p_n = 1 - \displaystyle\prod_{k=1}^{n-1} \left(1 - \frac{k}{365}\right)$
   (a) Ecrire une fonction appelée **anniversaire** prenant pour argument un entier naturel n et qui retourne la valeur p_n
   (b) Tester la fonction pour les entiers M, K et $M + K$ de votre identifiant
   (c) Déterminer (avec Python) l'entier n_0 tel que pour tout $n \geqslant n_0$, $p_n > 0.5$
   (d) Déterminer de même l'entier n_1 tel que pour tout $n \geqslant n_1$, $p_n > 0.9$

3. Sommes doubles

On rappelle que la somme double $\sum_{1 \leqslant i,j \leqslant 10} i\,j$ se calcule sous Python sous la forme :

```
s = 0
for i in range(1,11):
  for j in range(1,11):
    s = s + i*j
print(s)
```

(a) Ecrire une fonction appelée **SommeDouble** prenant pour argument un entier naturel n et qui retourne la valeur $S_n = \sum_{1 \leqslant i,j \leqslant n} \max(i\,j)$. (On pourra utiliser la fonction max de Python)

(b) Tester la fonction pour les entiers M, K et $M+K$ de votre identifiant

(c) Montrer (avec des tests Python) que pour tous les entiers n de votre liste personnelle **maliste**, on a $S_n = \dfrac{n(n+1)(4n-1)}{6}$. On essaiera de proposer un test qui ne repose pas sur un affichage de toutes les réponses pour les 100 valeurs à tester...

(d) Ecrire une fonction appelée **SommeTriangulaire** prenant pour argument un entier naturel n et qui retourne la valeur $T_n = \sum_{1 \leqslant i < j \leqslant n} \dfrac{i}{j}$.

(e) Tester la fonction pour les entiers M, K et $M+K$ de votre identifiant

(f) Montrer (avec des tests Python) que pour tous les entiers n de votre liste personnelle **maliste**, on a $T_n = \dfrac{n(n-1)}{4}$

Exercice 2 : Approximation de fonctions et tracé

Pour tout $x \in \mathbb{R}$ et $N \in \mathbb{N}$, on pose : $Aexp(N,x) = \sum_{k=0}^{N} \dfrac{x^n}{n!}$

1. Ecrire une fonction **Aexp** prenant en argument un entier N et un flottant x et qui retourne $Aexp(N,x)$. Vérifier que $Aexp(10,1)$ est une valeur approchée de e : quelle en est la précision ?

2. On rappelle que les commandes suivantes permettent de tracer les fonctions exp et sin sur le même graphe et sur l'intervalle $[-1,1]$:

```
import numpy as np
import matplotlib.pyplot as plt
x = np.linspace(-1,1,200)
plt.plot(x,np.sin(x))
plt.plot(x,np.exp(x))
plt.axhline()
plt.axvline()
plt.show()
```

Tracer sur le même graphe les fonctions $x \mapsto e^x$ et $x \mapsto Aexp(3,x)$ sur l'intervalle $[-2,2]$

3. Tracer sur le même graphe les fonctions $x \mapsto e^x$ et $x \mapsto Aexp(10,x)$ sur l'intervalle $[-2,2]$

Devoir d'informatique n° TP 1 (1,50 heures)

Ce TP noté est constitué de plusieurs petits exercices. Vous avez le droit d'utiliser les documents de cours ainsi que vos documents personnels mais, bien évidemment, pas ceux de vos voisins. Vous rendrez vos réponses sur la feuille de réponses jointe en y écrivant les valeurs ou réponses aux questions posées. Vous utiliserez Pyzo et effectuerez les imports de toutes les fonctions du module math ainsi que le module numpy avec l'alias np. De plus, pour vérifier le caractère personnel de votre travail, **vous enregistrerez tous les quarts d'heure** vos fichiers Python dans le dossier **Mes devoirs** d'où ils seront ramassés de manière régulière.

Attention : Pour les applications numériques ou d'autres questions, vous possédez chacun un identifiant personnel (M, K) couple de deux nombres donnés dans le tableau suivant :

Eleve	M	K		Eleve	M	K		Eleve	M	K
Bataillon	21	13		Devaux	21	14		Guillemin	21	15
Bouaouda	21	16		Breant	21	17		Raimand	21	18
Chadli	22	13		Chahboune	22	14		Dutriez	22	15
Bidault	22	16		Legrand	22	17		Paragot	22	18

Préliminaires : Utilisation de vos identifiants

Nous allons utilisez vos identifiants (M, K) pour ce TP, afin de déterminer une liste sur laquelle vous effectuerez certains tests des fonctions programmées.

Ecrire des séquences d'instructions permettant d'obtenir la liste **maliste** constituée des 100 entiers démarrant de M et séparés de K . Par exemple si on avait pris M = 10 et K = 2, on aurait eu la liste des 100 entiers pairs compris entre 10 et 208 . Sur la feuille des réponses, vous noterez le premier et le dernier terme de la liste **maliste**.

Exercice 1 : Travail avec les sommes et les produits

1. Somme des inverses des carrés des n premiers entiers $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2}$.
   (a) Ecrire une fonction appelée **zeta** prenant pour argument un entier naturel n et retourne la somme $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2}$.
   (b) Calculer **zeta**(N) où $N = M \times K$ est le produit de vos deux identifiants M et K
   (c) Trouver (avec Python) l'entier n_0 tel que pour tout $n \geqslant n_0$, $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2} > 1.5$
   (d) En déduire $\forall n \in \mathbb{N}^* \setminus \{1\}$, $\displaystyle\sum_{k=1}^{n} \frac{1}{k^2} > \frac{3n}{2n+1}$
2. Paradoxe des anniversaires
   La probabilité p_n qu'au moins deux étudiants d'une classe de $n \geqslant 2$ étudiants aient leur anniversaire le même jour de l'année est donnée par la formule : $p_n = 1 - \displaystyle\prod_{k=1}^{n-1} \left(1 - \frac{k}{365}\right)$
   (a) Ecrire une fonction appelée **anniversaire** prenant pour argument un entier naturel n et qui retourne la valeur p_n
   (b) Tester la fonction pour les entiers M, K et $M + K$ de votre identifiant
   (c) Déterminer (avec Python) l'entier n_0 tel que pour tout $n \geqslant n_0$, $p_n > 0.5$
   (d) Déterminer de même l'entier n_1 tel que pour tout $n \geqslant n_1$, $p_n > 0.9$

3. Sommes doubles

On rappelle que la somme double $\sum_{1 \leqslant i,j \leqslant 10} i\,j$ se calcule sous Python sous la forme :

```
s = 0
for i in range(1,11):
  for j in range(1,11):
    s = s + i*j
print(s)
```

(a) Ecrire une fonction appelée **SommeDouble** prenant pour argument un entier naturel n et qui retourne la valeur $S_n = \sum_{1 \leqslant i,j \leqslant n} \max(i\,j)$. (On pourra utiliser la fonction max de Python)

(b) Tester la fonction pour les entiers M, K et $M + K$ de votre identifiant

(c) Montrer (avec des tests Python) que pour tous les entiers n de votre liste personnelle **maliste**, on a $S_n = \dfrac{n(n+1)(4n-1)}{6}$. On essaiera de proposer un test qui ne repose pas sur un affichage de toutes les réponses pour les 100 valeurs à tester...

(d) Ecrire une fonction appelée **SommeTriangulaire** prenant pour argument un entier naturel n et qui retourne la valeur $T_n = \sum_{1 \leqslant i < j \leqslant n} \dfrac{i}{j}$.

(e) Tester la fonction pour les entiers M, K et $M + K$ de votre identifiant

(f) Montrer (avec des tests Python) que pour tous les entiers n de votre liste personnelle **maliste**, on a $T_n = \dfrac{n(n-1)}{4}$

Exercice 2 : Approximation de fonctions et tracé

Pour tout $x \in \mathbb{R}$ et $N \in \mathbb{N}$, on pose : $Aexp(N, x) = \sum_{k=0}^{N} \dfrac{x^n}{n!}$

1. Ecrire une fonction **Aexp** prenant en argument un entier N et un flottant x et qui retourne $Aexp(N, x)$. Vérifier que $Aexp(10, 1)$ est une valeur approchée de e : quelle en est la précision ?

2. On rappelle que les commandes suivantes permettent de tracer les fonctions exp et sin sur le même graphe et sur l'intervalle $[-1, 1]$:

```
import numpy as np
import matplotlib.pyplot as plt
x = np.linspace(-1,1,200)
plt.plot(x,np.sin(x))
plt.plot(x,np.exp(x))
plt.axhline()
plt.axvline()
plt.show()
```

Tracer sur le même graphe les fonctions $x \mapsto e^x$ et $x \mapsto Aexp(3, x)$ sur l'intervalle $[-2, 2]$

3. Tracer sur le même graphe les fonctions $x \mapsto e^x$ et $x \mapsto Aexp(10, x)$ sur l'intervalle $[-2, 2]$

Feuille de réponses Nom : .... Prénom : ....

Préliminaire : Utilisation de vos identifiants

1. Noter le premier et le dernier terme de la liste **maliste**

Exercice 1 : Travail sur les sommes et les produits

1. Somme des inverses des carrés des n premiers entiers
   - Notez $\zeta(N)$ avec 7 chiffres significatifs
   - Notez n_0
   - Donnez la démonstration demandée
2. Paradoxe des anniversaires
   - Notez anniversaire(M), anniversaire(K) et anniversaire(M+K)
   - Notez n_0
   - Notez n_1
3. Sommes doubles
   - Notez SommeDouble(M), SommeDouble(K) et SommeDouble(M+K)
   - Notez SommeTriangulaire(M), SommeTriangulaire(K) et SommeTriangulaire(M+K)
   - Notez n_1

Feuille de réponses Nom : .... Prénom : ....

Préliminaire : Utilisation de vos identifiants

1. Noter le premier et le dernier terme de la liste **maliste**

Exercice 1 : Triplets pythagoriciens

- Notez Triplets(M), Triplets(K) et Triplets(N)

- Notez TRPI(N)

Exercice 2 : Travail sur les sommes et les produits

1. Somme des inverses des carrés des n premiers entiers
   - Notez $\zeta(N)$ avec 7 chiffres significatifs
   - Notez n_0
   - Donnez la démonstration demandée
2. Paradoxe des anniversaires
   - Notez anniversaire(M), anniversaire(K) et anniversaire(M+K)
   - Notez n_0
   - Notez n_1
3. Sommes doubles
   - Notez SommeDouble(M), SommeDouble(K) et SommeDouble(M+K)
   - Notez SommeTriangulaire(M), SommeTriangulaire(K) et SommeTriangulaire(M+K)
   - Notez n_1

Exercice 1 : Un peu d'arithmétique

1. Triangles rectangles. On appelle triplet pythagoricien tout triplet (α, β, γ) d'entiers tels que que $0 < \alpha \leqslant \beta \leqslant \gamma$ et $\alpha^2 + \beta^2 = \gamma^2$

   (a) Ecrire une fonction **Paire** prenant comme argument deux entiers a et b et qui renvoie le booléen `True` si (a, b, c) avec $c = \sqrt{a^2 + b^2}$ est un triplet pythagoricien et `False` sinon.

   (b) Ecrire une fonction **Triplets** prenant comme argument un entier naturel n et qui retourne tous les triplets pythagoriciens (a, b, c) avec $0 < a \leqslant b \leqslant n$

   (c) Faites le test avec n valant la somme de vos identifiants M et K. De même, combien y-a-t-il de triplets pythagoriciens (a, b, c) avec $0 < a \leqslant b \leqslant N$ où est le produit de vos identifiants M et K ?

   (d) En utilisant la fonction **Paire** , écrire une fonction **TRPI** prenant comme argument un entier naturel n et qui retourne tous les triplets pythagoriciens de la forme $(a, a+1, c)$ avec $0 < a < n$.

   (e) Tester cette fonction avec $N = M \times K$ où M et K sont vos identifiants.

2. Nombres parfaits et nombres amicaux

   (a) Ecrire une fonction **SommeDiviseur** prenant comme argument un entier naturel n et qui retourne la somme des diviseurs stricts de n (c'est-à-dire les diviseurs positifs de n sauf n lui-même). Faites le test avec la dernier terme de votre liste **maliste**.

   (b) Ecrire une fonction **parfait** prenant comme argument un entier naturel n et qui retourne le booléen `True` si n vaut la somme de ses diviseurs stricts et `False` sinon. Trouver tous les nombres parfaits inférieurs à $N = M \times K$ où M et K sont vos identifiants.

   (c) Ecrire une fonction **ami** prenant comme argument un entier naturel n et qui retourne le booléen `True` s'il existe un entier m tel que la somme des diviseurs stricts de n soit m et la somme des diviseurs stricts de m soit n. n vaut la somme de ses diviseurs stricts et `False` sinon. Trouver tous les couples (n, m) avec $n < m \leqslant N$ avec $N = M \times K$ où M et K sont vos identifiants.