

Exercice 1 (Des suites récurrentes)

1. Écrire une fonction `suites(n, a, b)` qui renvoie une liste avec deux éléments où le premier élément est a_n et le second élément est b_n où on pose $a_0 = a, b_0 = b \in \mathbb{R}^+$ avec $b \geq a$ et pour tout $n \in \mathbb{N}$:

$$a_{n+1} = \sqrt{a_n b_n} \quad \text{et} \quad b_{n+1} = \frac{a_n + b_n}{2}.$$

2. Réécrire la fonction précédente pour que celle-ci retourne le message '`a <= b est faux`' lorsque $a \leq b$ n'est pas vérifié.
3. Écrire une fonction `test(a,b,N)` qui teste si $a_i \leq b_i$ pour tout entier $i \leq N$. Elle renverra '`a_i inferieure ou egal a b_i pour tout i inferieur ou egal a N`' si ceci est vrai et elle renverra '`l inegalite est fausse au rang k`' où k est le rang auquel l'inégalité est fausse.
4. On admet que pour tout $n \in \mathbb{N}, a_n \leq b_n$. Montrer que (a_n) est croissante et (b_n) décroissante.

Exercice 1 :

Question 1 :

Question 2 :

Question 3 :

Question 4 : Montrer que (a_n) est croissante et (b_n) décroissante.

Exercice 2 (*Des comparaisons de suites*)

On rappelle que pour obtenir la valeur absolue, la commande Python est `abs(a)` où a est le nombre dont on veut obtenir la valeur absolue et que `exp(1)` permet d'avoir la valeur de e .

1. Écrire une fonction `facto_pair(n)` qui renvoie $2 \times 4 \times \dots \times (2n)$ pour un entier n .
2. Réécrire la fonction `facto_pair(n)` précédente et qui renverra une erreur `TypeError` lorsque n n'est pas un entier avec le message « 1 argument n est pas un entier ».
3. Écrire une fonction `suite(n)` qui renvoie $v_n = \sqrt{2\pi n} \left(\frac{2n}{e}\right)^n$.
4. Écrire une fonction `repres(n)` qui représente graphiquement les termes u_1, \dots, u_n où pour tout entier $k \in \mathbb{N}^*$:

$$u_k = \frac{2 \times 4 \times \dots \times (2k)}{v_k}.$$

5. Écrire une fonction `plus_petit_element(P)` qui retourne le plus petit entier n tel que

$$|u_k - 1| \leq P.$$

Exercice 2 :

Question 1 :

Question 2 :

Question 3 :

Question 4 :

Question 5 :

Exercice 3 (*Diviseurs*)

On rappelle que la commande Python `a%b` renvoie la reste de la division euclidienne de a par b et que b est un diviseurs de a lorsque ce reste est nul.

1. Écrire une fonction `diviseurs(n)` qui renvoie la liste des diviseurs de n .
2. On donne la fonction `mystere(a,b)` plus bas. Expliquer le rôle de cette fonction puis réécrire cette fonction en créant une documentation.
3. À l'aide de la question 1 ou de la question 2, écrire une fonction `PGCD(a,b)` qui retourne le plus grand diviseur commun de a et b .
4. Convertir le nombre $\overline{100010}^2$ en écriture et $\overline{245}^{10}$ en binaire. On expliquera dans chaque cas la méthode employée.
5. Écrire une fonction `dec_i_bin(a)` qui à partir d'un nombre a écrit en décimal retourne une liste correspondant à l'écriture en binaire de a . On pourra utiliser la commande `L.insert(0,i)` qui insère en première position l'élément i dans la liste L .

Exercice 3 :

Question 1 :

Question 2 : `def mystere(a,b):`

```
L=[]
for i in range(1,min(a,b)+1):
    if a%i == 0 and b%i == 0:
        L.append(i)
    else:
        pass
return L
```

Explications :

Réécriture de la fonction : `def mystere(a,b):`

Question 3 : Convertir le nombre $\overline{100010}^2$ en écriture et $\overline{245}^{10}$ en binaire.

Question 4 :

