Math93.com

TD n°4 - Algorithmes
Suites et problèmes (historiques)

Histoire
De l'approche historique des problèmes. Le problème de Bâle, la suite de Babylone ...

Exercice 1. Le problème de Bâle (déjà vu dans les TD sur les suites n°1 et n°2)

Soit pour tout entier $n \geq 1$, la suite (u_n) définie par :

$$u_n = \sum_{k=1}^{n} \frac{1}{k^2}$$

On a montré que cette suite était convergente.
L'objectif de cet exercice est de valider les affirmation de la remarque historique.

Remarque historique

En mathématiques, le problème de Bâle (ou problème de Mengoli) est un problème qui consiste à demander la valeur de la somme de la série des inverses de carrés des entiers (non nuls).
Le problème a été résolu par le génial mathématicien suisse Leonhard Euler (1707-1783) qui parvient à démontrer que cette somme tend vers $\frac{\pi^2}{6}$. Il en donna la première démonstration rigoureuse en 1741 mais annonce en 1735 la découverte de la somme exacte.

$$1+\frac{1}{2^2}+\frac{1}{3^2}+\frac{1}{4^2}+\frac{1}{5^2}+\cdots=\sum_{k=1}^{+\infty}\frac{1}{k^2}=\frac{\pi^2}{6}$$

Pour obtenir 4 décimales exactes, il faut additionner plus de 15 000 termes de la somme. Avec 1000 termes, on n'obtient que 2 décimales et la fraction irréductible comporte déjà plus de 800 chiffres. Cela reste rêveur quand on pense qu'Euler a calculé 20 décimales exactes (mais avec des méthodes d'accélération de convergence).

1. Calculer les 5 premiers termes de la suites.
2. Écrire une fonction $u(n)$ qui prend en entrée un entier n et qui détermine puis renvoie le terme u_n de la suite.

```
def u(n):
    '''In : indice n, entier naturel
      Out : u(n)'''
    assert n>=1
    ...
    return ...
```

3. Écrire une fonction $f(n)$ qui prend en entrée un entier n et qui détermine puis renvoie l'écart entre le terme u_n de la suite et $\frac{\pi^2}{6}$.
Attention il faut importer le module math pour pouvoir utiliser une valeur approchée de π.

```
from math import *
def f(n):
    '''In : indice n, entier naturel
      Out : u(n) - pi**2 /6'''
    assert n>=1
    ...
    return ...
```

4. Écrire une fonction $g(p)$ qui prend en entrée un entier n et qui détermine puis renvoie le rang n à partir duquel l'écart entre le terme u_n de la suite et $\frac{\pi^2}{6}$ est inférieur à 10^{-p}.
Vérifier alors les assertions de la remarque historique.

Exercice 2. Suite de « Babylone » (vu dans le TD 2 sur les suites)

Remarque historique

En mathématiques, **la méthode de Héron** ou méthode babylonienne est une méthode efficace d'extraction de racine carrée, c'est-à-dire de résolution de l'équation $x^2 = a$, avec a positif.
Elle porte le nom du mathématicien **Héron d'Alexandrie (1er siècle)**, qui l'expose dans le tome I de son ouvrage *Metrica* (Les métriques). On estime que les babyloniens et égyptiens connaissaient déjà cette méthode vers -2000.

Pour déterminer la racine carré d'un nombre positif a, on choisit u_0 assez proche de $\sqrt{a}$ puis on définit la suite

$$\begin{cases} u_0 \\ u_{n+1} = \dfrac{1}{2}\left(u_n + \dfrac{a}{u_n}\right) \end{cases}$$

Du fait de sa convergence rapide, la méthode de Héron permet d'obtenir une bonne approximation de la valeur de $\sqrt{a}$ même après peu d'étapes de calcul.

Par exemple avec $a = 2$, on constate que la convergence est **quadratique** (2 décimales exactes au deuxième calcul, 5 au troisième, 11 au quatrième, 23 au cinquième).
En seulement trois étapes, la précision relative sur la valeur de $\sqrt{2}$ est déjà de 10^{-6}, ce qui est excellent, et de moins de 10^{-12} en quatre étapes.
De fait, une des principales problématiques est de choisir une « bonne » valeur pour u_0, idéalement l'entier dont le carré est le plus proche de a, ce que suggérait d'ailleurs Héron lui-même dans la partie des *Metrica* consacrée à cette question.

On considère la suite (u_n) définie par :

$$\begin{cases} u_0 = b \\ u_{n+1} = \dfrac{1}{2}\left(u_n + \dfrac{a}{u_n}\right) \end{cases}$$

1. On prend $a = 2$ et $u_0 = b = 1$. Calculer les 5 premiers termes de la suites.
2. Écrire une fonction $u(n, a, b)$ qui prend en entrée un entier n, deux nombres positifs a et b, et qui détermine puis renvoie le terme u_n de la suite (u_n).

```python
def u(n,a,b):
    '''In : indice n, entier naturel
      Out : u(n)'''
    assert n>=0   and a>=0 and b>=0
    ...
    return ...
```

3. Écrire une fonction $f(n, a, b)$ qui prend en entrée un entier n, deux nombres positifs a et b, et qui détermine puis renvoie l'écart entre le terme u_n de la suite et $\sqrt{a}$.

module math

*Attention il faut importer le **module math** pour pouvoir utiliser une valeur approchée de π (avec **pi**). Par contre la **valeur absolue** se note **abs()** et est nativement accessible.*

```python
from math import * # pour utiliser pi
def f(n,a,b):
    '''In : indice n, entier naturel
      Out : abs(u(n) - racine(a))'''
    assert n>=0   and a>=0 and b>=0
    ...
    return ...
```

4. Écrire une fonction $g(p, a, b)$ qui prend en entrée un entier n, deux nombres positifs a et b, et qui détermine puis renvoie le rang n à partir duquel l'écart entre le terme u_n de la suite et $\sqrt{a}$ est inférieur à 10^{-p}.
   Vérifier alors les assertions de la remarque historique.

Fin du devoir