Devoir à la Maison n°1

Calculs de π

Le premier but de ce devoir est de calculer π avec une précision correcte. Nous chercherons ensuite la meilleure approximation rationnelle possible de π.

Le langage utilisé doit être Python.

Il est inutile au sein d'une fonction de vérifier que les variables d'entrée satisfont les conditions nécessaires pour que le calcul soit défini.

Par contre, il est important de faire figurer la documentation des fonctions, et de commenter les lignes de code.

Les tests sont explicitement demandés dans certaines questions, sinon il n'est pas nécessaire de les faire figurer sur la copie.

Partie A. Méthode de Monte-Carlo

Soit $\mathcal{P}$ le pavé $[0,1]\times[0,1]$, c'est-à-dire l'ensemble des points M dont les coordonnées (x,y) sont toutes les deux comprises dans l'intervalle $[0,1]$.
Un point $M(x,y)$ de ce pavé est à l'intérieur du cercle trigonométrique si et seulement si $x^2+y^2\leqslant 1$.
Si x et y sont choisis aléatoirement et uniformément dans l'intervalle $[0,1]$ alors la probabilité que M soit à l'intérieur du cercle trigonométrique est $\frac{\pi}{4}$: c'est la surface du quart de disque trigonométrique divisée par la surface du pavé.
Soit N un grand nombre. On choisit N points au hasard dans le pavé et on compte la proportion de ces points appartenant au quart de disque. On obtient alors une approximation de $\frac{\pi}{4}$, puis, en la multipliant par 4, on obtient une approximation de π.
La fonction `random` du module de même nom renvoie un réel aléatoire de l'intervalle $[0,1]$, uniformément choisi. On importe cette fonction en plaçant au début du programme :

```python
from random import random
```

On l'utilise de la façon suivante :

```python
>>> x=random()
>>> print(x)
```

L'algorithme à mettre en œuvre est le suivant :

- demander à l'utilisateur un entier N qui sera le nombre de points à tester
- effectuer N fois l'opération consistant à tirer un point (donc deux coordonnées) au hasard, déterminer si ce point est à l'intérieur du cercle trigonométrique, et si c'est le cas incrémenter un compteur,
- afficher l'approximation de π obtenue.

1. Écrire un programme qui calcule une approximation de π en utilisant la méthode de Monte-Carlo.

2. Tester ce programme pour de grandes valeurs de N.
Indiquer votre meilleur résultat.

On souhaite illustrer graphiquement ce programme. Pour cela on importe au début du programme le module permettant de tracer des graphes :

```
from matplotlib.pyplot import *
```

Pour placer un point de coordonnées (x, y) :

```
plot([x],[y],'b+')
```

Le `'b'` demande que le point soit affiché en bleu. On peut aussi utiliser le rouge (`'r'`), le vert (`'g'`), le noir (`'k'`), etc.

Le `'+'` demande que le point soit représenté par un $+$, on peut le remplacer par un cercle (`'o'`), une croix (`'x'`), etc.

À la fin du programme on ajoutera les instructions :

```
# Affichage des axes :
axhline(color='k')
axvline(color='k')

# Repère orthonormé :
axis("equal")

# Fenêtre à afficher :
axis([0,1,0,1])

# Affichage :
show()
```

La dernière ligne est indipensable, ce n'est que là que le graphique est affiché à l'écran.

3. Ajouter au programme l'affichage des N points.

Les points à l'intérieur du cercle trigonométrique seront affichés dans une couleur, ceux à l'extérieur dans une autre couleur.

4. Tester pour une valeur de N assez grande. Joindre une image du résultat.

Partie B. La fonction arc-tangente

La fonction arc-tangente, notée arctan, est définie sur $\mathbb{R}$. À tout réel x elle associe l'angle θ de l'intervalle $\left]-\frac{\pi}{2}, \frac{\pi}{2}\right[$ tel que $\tan\theta = x$. Ainsi : $\arctan 1 = \frac{\pi}{4}$.
Nous admettons que :

$$\forall x \in [-1,1], \qquad \arctan x = \lim_{n\to+\infty} S_n(x), \qquad \text{où} \qquad S_n(x) = \sum_{k=0}^{n} (-1)^k \frac{x^{2k+1}}{2k+1}.$$

Ainsi, si $\frac{x^{2n+1}}{2n+1}$ est très petit alors $S_n(x)$ est une bonne approximation de $\arctan x$.

5. Écrire une fonction qui reçoit $n \in \mathbb{N}$ et $x \in \mathbb{R}$ et qui calcule $S_n(x)$.

6. Écrire une fonction qui reçoit un réel x de l'intervalle $[-1,1]$ et un réel strictement positif ε, et qui renvoie $S_n(x)$ où n est un entier tel que $\left|\frac{x^{2n+1}}{2n+1}\right|$ est inférieur à ε.

Cette fonction ressemble à la précédente, si ce n'est qu'elle utilise une boucle **while**. On admet qu'elle donne une approximation de $\arctan x$ avec une précision de ε.

7. Sachant que $\arctan 1 = \frac{\pi}{4}$, donner une approximation de π avec une précision de 10^{-6}. Peut-on faire mieux en temps raisonnable ?

La formule de Machin (du mathématicien anglais John Machin, 1680 – 1751) s'écrit :

$$\frac{\pi}{4} = 4\arctan\frac{1}{5} - \arctan\frac{1}{239}\cdot$$

8. Utiliser cette formule pour déterminer la valeur de π avec une précision de 10^{-16}.

Partie C. Approximation rationnelle de π

Dans cette partie on s'autorise l'importation de la fonction sinus et de la constante π depuis le module `math`, ceci grâce à l'instruction suivante en début de programme :

```
from math import sin,pi
```

On cherche une bonne approximation de π par un rationnel, c'est-à-dire deux entiers p et q tels que $\pi \simeq \frac{p}{q}$. Ou souhaite que l'approximation soit bonne mais que p et q ne soient pas trop grands.

Par exemple, l'approximation $\pi \simeq \frac{22}{7}$ est connue depuis l'antiquité. Elle est correcte à 10^{-2} près : $\frac{22}{7} = 3{,}1428...$

Le principe de l'algorithme est le suivant : on calcule $\sin(p)$ pour tous les entiers p allant de 1 à N, où N est choisi par l'utilisateur.

Si $\sin(p)$ est proche de 0 alors il existe $q \in \mathbb{Z}$ tel que $p \simeq q\pi$, ce qui donne $\pi \simeq \frac{p}{q}$.

On cherche donc l'entier p compris entre 1 et N tel que $|\sin(p)|$ est minimal. Ensuite on calcule $q \simeq \frac{\pi}{p}$ grâce à l'instruction **round** qui arrondit un réel à l'entier le plus proche.

9. Écrire un programme qui demande N à l'utilisateur, puis qui affiche $|\sin p|$ pour tout p allant de 1 à N.

10. Modifier le programme précédent pour déterminer p tel que $|\sin p|$ est minimal. Ajouter le calcul de q puis l'affichage de p et de q, ainsi que la différence entre $\frac{p}{q}$ et π.

11. Déterminer une approximation de π à 10^{-6} près avec q minimal. Quelle est la meilleure approximation suivante ?