Devoir maison 7

A rendre le au plus tard.

Exercice 1
Le but de cet exercice est d'étudier une manière d'approximer π par des fractions.

Partie I

On souhaite prouver que $\frac{\pi}{4} = 4\arctan(\frac{1}{5}) - \arctan(\frac{1}{239})$ (Formule de Machin).

1. On pose $f : \left\{ \begin{array}{ccc}]-1,1[& \to & \mathbb{R} \\ x & \mapsto & 2\arctan(x) - \arctan(\frac{2x}{1-x^2}) \end{array} \right.$. Montrer que f est dérivable sur $]-1,1[$ et constante sur cet intervalle. Préciser la valeur de la constante.
2. En déduire $4\arctan\frac{1}{5}$.
3. Soient $a, b \in]-1,1[$. Montrer que $\arctan(a) + \arctan(b) = \arctan(\frac{a+b}{1-ab})$ en posant $a = \tan(\alpha)$ et $b = \tan(\beta)$ (on précisera le choix de α et β).
   Que peut-on dire de $\arctan(a) - \arctan(b)$?
4. En déduire la formule de Machin. On pourra éventuellement utiliser Python pour cette question. Voir l'exercice 2.

Partie II

Passons à la pratique. Nous allons essayer de déduire une valeur approchée de π de cette formule en calculant des valeurs approchées de arctan.

1. On pose $f_1 : \left\{ \begin{array}{ccc} \mathbb{R}^+ & \to & \mathbb{R} \\ x & \mapsto & \arctan(x) - x + \frac{x^3}{3} \end{array} \right.$.
   (a) Montrer que $\forall x \in \mathbb{R}^+ \ 0 \leqslant f_1(x) \leqslant \frac{x^5}{5}$.
   (b) En déduire un encadrement de arctan valable sur $\mathbb{R}^+$.
2. Soit $n \in \mathbb{N}$. Posons $f_n : x \mapsto \arctan(x) - \sum_{k=0}^{n} \frac{(-1)^k}{2k+1} x^{2k+1}$. Donner le signe de f_n sur $\mathbb{R}^+$ après avoir étudier ses variations. Indication : la parité de n joue un jeu, et la somme n'a pas cette valeur par hasard.
3. En déduire que $\forall x \in \mathbb{R}^+ \ |f_n(x)| \leqslant \frac{x^{2n+3}}{2n+3}$. On pourra s'intéresser à la fonction f_{n+1}.
4. Nous avons démontré que pour $x \in \mathbb{R}^+$ fixé, $\mathrm{P}_n(x) = \sum_{k=0}^{n} \frac{(-1)^k}{2k+1} x^{2k+1}$ est une approximation de $\arctan(x)$ à $\frac{x^{2n+3}}{2n+3}$ près. Montrer que $\mathrm{P}_n(\frac{1}{5})$ est une approximation de $\arctan(\frac{1}{5})$ à 10^{-p} ($p \in \mathbb{N}$ fixé) ssi $\ln(2n+3) + (2n+3)\ln(5) \geqslant p\ln(10)$.
   Que devient cette condition pour $\mathrm{P}_n(\frac{1}{239})$ (pour la même approximation) ?
5. Supposons qu'on ait deux entiers m, n tels que $-10^{-p} \leqslant f_n(\frac{1}{5}) \leqslant 10^{-p}$ et $-10^{-p} \leqslant f_m(\frac{1}{239}) \leqslant 10^{-p}$. Encadrer $\pi - (16\mathrm{P}_n(\frac{1}{5}) - 4\mathrm{P}_m(\frac{1}{239}))$. Remarquer que l'erreur commise devient plus grande après les opérations arithmétiques.

Exercice 2
Passons au Python ! Nous allons convertir cette étude en programme permettant d'obtenir de multiples décimales de π.

Fractions

En python, pour manipuler les fractions on utilise la commande suivante

```python
from fractions import Fraction  # seulement une partie du module fractions
# et pour créer des fractions par la suite
f = Fraction(num, denom)
```

où num est le numérateur et dénom le dénominateur. Une fois la fraction f créée on accède à ses numérateur et dénominateur par

```python
f.numerator
f.denominator
```

Code Créer une fonction **developpement(f, n)** qui retourne la chaîne de caractères correspondant au développement décimal (nombre à virgule) de la fraction f tronqué à n chiffres derrière la virgule. Pour trouver l'algorithme, il est important de traiter quelques exemples à la main.
Outil : pour transformer un nombre en chaîne, on utilisera la fonction str.

Polynômes

Nous allons devoir calculer $\mathrm{P}_n(x) = \sum_{k=0}^{n} (-1)^k \frac{x^{2k+1}}{2k+1}$ pour différents x et n.

Code Créer la fonction **pn(n, x)** qui renvoie cette somme, sans faire appel à la fonction **sum** de python.

Raffinement Le code précédent est très peu efficace. En particulier on ne réutilise pas les calculs de puissances déjà effectués. Nous allons implémenter l'algorithme dit de Horner. Si $f : x \mapsto \sum_{k=0}^{n} a_k x^k$ est une fonction polynomiale alors

$$f(x) = (\dots(((0 + a_n)x + a_{n-1})x + a_{n-2})x \cdots + a_1)x + a_0.$$

L'algorithme consiste donc à faire les calculs à l'envers : partir de a_n, le multiplier par x, ajouter le coefficient précédent, puis multiplier le tout par x et recommencer.

Code Créer une fonction **horner(n, x)** qui à la même valeur de retour que **pn** mais utilise l'algorithme précédent.
Outil : pour obtenir un range décroissant on peut utiliser reversed(range()) ou range(valeurmax, -1, -1)

π !

Code Utiliser l'étude mathématique pour remplir les fonctions **trouve_n** et **trouve_m**
Remarquons que le premier exercice nous permet d'affirmer que si n est tel que $|f_n(\frac{1}{5})| \leqslant 10^{-p-2}$ et m est tel que $|f_m(\frac{1}{239})| \leqslant 10^{-p-1}$ alors $16\mathrm{P}_n(\frac{1}{5}) - 4\mathrm{P}_m(\frac{1}{239})$ est une approximation de π à 10^{-p} près.

Code Remplir la fonction **approxime_pi**
Il ne vous reste plus qu'à remplacer la variable mon_pi par une fraction qui est une approximation de π à 10^{-100} près. Utilisez **developpement** pour admirer toutes ces décimales de π que nous venons de calculer.