Rapport de projet d'analyse
Certification des n premières décimales de $\sin(\frac{8}{5})$

Mostafa HAGHVIRDILOO
Marion NOVERRAZ
Alexandre RAMDOO
Henri MACEDO GONCALVES

May 13, 2022

Sommaire

1 Introduction

Notre projet a pour but d'établir avec certitude la partie entière ainsi que les 6 premières décimales de $\sin(\frac{8}{5})$. Ainsi, nous utiliserons les théorèmes de Taylor et de Cauchy pour pouvoir établir les fondements mathématiques de notre raisonnement.

2 Motivation de l'équipe

2.1 Choix du sujet

Nous avons tout de suite été intéressé par ce projet car le concept de code certifiant $\sin(\frac{8}{5})$ est un concept qui nous a tous intrigué. D'autre part, établir une suite de rationnels $(r_n)_{n\in\mathbb{N}}$ telle que $\lim\limits_{n\to+\infty} r_n = a$ nous permettrait d'utiliser et de mobiliser nos connaissances, en réalisant un projet basé essentiellement sur l'analyse et l'algorithmique.

2.2 Travail supplémentaire

Après avoir répondu aux questions du sujet, nous avons souhaité rendre nos résultats accessibles et utiles. Ainsi, convappro, le module que nous avons créé, peut-être installé par tout utilisateur de python souhaitant utiliser simplement le sinus d'un réel(en radians compris entre 0 et 2π) avec une précision théorique de l'ordre de 10^{-100}. Les informations relatives à ce module sont disponibles sur sa page PyPi: `https://pypi.org/project/convappro/`

3 Le projet

Nous avons décidé de travailler de manière collaborative à l'élaboration du code et aussi pour la rédaction du rapport. Pour cela, nous avons créé un répertoire Github. Nous avons donc pu réaliser plus facilement des bilans de nos différentes recherches à plusieurs reprises.

3.1 PARTIE A

3.1.1 Construction d'une suite de rationnels $(r_n)_{n\in\mathbb{N}}$ telle que $\lim\limits_{n\to+\infty} (r_n)_{n\in\mathbb{N}} = \mathbf{a} = \sin(\frac{8}{5})$

Nous allons construire une suite de rationnels $(r_n)_{n\in\mathbb{N}}$ telle que $\lim\limits_{n\to+\infty} r_n = a$. On en explicitera ensuite un certificat de convergence.

La formule de Taylor (théorème 3.9.1) donne:

Soit f une fonction de classe $C^k, k \in \mathbb{N}$ définie sur I, supposons que $f^{(k+1)}$ est bornée. Alors f admet un développement limité, à l'ordre k en 0 et:

$\forall h \in I, f(h) = \sum_{n=0}^{k} \frac{f^{(n)}(0)}{n!} h^n + h^{k+1}\varepsilon(h)$

Avec : $sup|\varepsilon(h)| \leq sup\frac{|(f)^{k+1}(x)|}{(k+1)!}$.

On pose ensuite f:= sin.

Alors:
f':= cos
f":= -sin
f$^{(3)}$:= -cos
f$^{(4)}$:= sin

etc..

Or $\begin{cases} cos(0) = 1 \\ sin(0) = 0 \end{cases}$

D'où:$\begin{cases} f^{(n)}(0) = 1 \text{ ou } -1 \text{ si n impair} \\ f^{(n)}(0) = 0 \text{ sinon} \end{cases}$

Ainsi, on a:
$\forall k \in \mathbb{N}, \forall h \in I, f(h) = (\sum_{n=0}^{k} \frac{(-1)^n h^{2n+1}}{(2n+1)!}) + o(h^{k+1})$ qui est la formule de Taylor pour le calcul du sinus. Et comme $\lim\limits_{k\to+\infty} o(h^{k+1}) = \lim\limits_{k\to+\infty} h^{k+1}\varepsilon(h) = 0$, on en déduit donc que la suite $(r_n)_{n\in\mathbb{N}}$ que nous recherchons est donc définie par f(h) avec k := n et h = $\frac{8}{5}$.

$r_n : \mathbb{N} \to \mathbb{R}$

$n \to \sum_{k=0}^{n} \frac{(-1)^k(\frac{8}{5})^{2k+1}}{(2k+1)!}$

Et comme $\sin(\frac{8}{5}) = \sum_{k=0}^{n} \frac{(-1)^k(\frac{8}{5})^{2k+1}}{(2k+1)!} + (\frac{8}{5})^{n+1}\varepsilon(\frac{8}{5})$, on certifie que $\lim\limits_{n\to+\infty} r_n = sin(\frac{8}{5}) = a$
Car $\lim\limits_{n\to+\infty} (\frac{8}{5})^{n+1}\varepsilon(\frac{8}{5}) = 0$, (d'après la formule de Taylor).

On vient alors de construire une suite $(r_n)_{n\in\mathbb{N}}$ telle que

$$\lim_{n\to+\infty} r_n = a$$

De plus, on a montré que $(r_n)_{n\in\mathbb{N}}$ converge vers a. On peut donc établir un certificat de convergence :

D'après ce que nous avons vu plus haut :
$\forall h \in I, f(h) = \sum_{n=0}^{k} \frac{f^{(n)}(0)}{n!} h^n + h^{k+1}\varepsilon(h)$ avec : $sup|\varepsilon(h)| \leq sup\frac{|(f)^{k+1}(x)|}{(k+1)!}$.
et avec f:= sin $\begin{cases} f^{(n)}(0) = 1 \text{ ou } -1 \text{ si n impair} \\ f^{(n)}(0) = 0 \text{ sinon} \end{cases}$
donc $sup|\varepsilon(h)| \leq sup\frac{|(f)^{k+1}(x)|}{(k+1)!} \leq \frac{1}{(k+1)!}$

d'où on peut fournir un certificat du type :
conv: $\mathbb{N} \to \mathbb{R}$
$k \mapsto conv(k)$
avec $conv(k) = \frac{1}{(n+1)!}$ qui garantit la stabilité de l'écriture de r(n) jusqu'à la k-ième décimale dès que $n \geq conv(k)$. En des termes plus formels : $\forall k \in \mathbb{N}, \forall n \geq conv(k), |r(n) - r(conv(k))| < 10^{-k}$.

3.2 PARTIE B

3.2.1 Écriture d'une fonction Python r(n) qui renvoie un flottant approchant $(r_n)_{n\in\mathbb{N}}$ (à la précision près de l'arithmétique du CPU).

La fonction reprend la formule de Taylor à tout ordre de précision p souhaité en effectuant le dévéloppement limité à l'ordre désiré sans pour autant perdre en précision. Pour cela, nous avons utilisé des fonctions auxiliaires notamment de troncature et de calcul factoriel.

3.2.2 Écriture du certificat de convergence de $(r_n)_{n\in\mathbb{N}}$ à 10^{-k} près conv(k)

La fonction conv(k) calcule à la précision près du CPU $(r_n)_{n\in\mathbb{N}}$ pour en établir et retourner un ordre k du certificat de convergence. De plus, nous avons implémenté la fonction PreuveCauchy utilisant les suites et la preuve de Cauchy pour établir un ordre de grandeur de la précision souhaitée soit l'ordre minimal de $(r_n)_{n\in\mathbb{N}}$ à atteindre pour en obtenir un résultat certifiable à 10^{-p} près.

3.2.3 Affichage des résultats

Pour rendre nos résultats modulables, nous avons decidé d'adopter une structure orientée objet. L'objet de classe Calculs est ainsi initié par la valeur flottante sur laquelle porte notre étude (ici 8/5). De cette manière, nous définissons l'instance outils de la classe Calculs que nous utiliserons tout le long de notre analyse. Sur cette instance, nous avons effectué les opérations et affichages demandés pour parvenir aux résultats attendus.

4 Conclusion

En usant d'un tel niveau de modularité lors de l'écriture de notre code, nous espérons le rendre capable, par extension, de certifier avec le même niveau de précision le sinus de n'importe quel autre flottant en utilisant les méthodes décrites plus haut, elles-mêmes implémentées dans la classe Calculs.