

Devoir confiné

Comment utiliser Scilab depuis chez soi ? (si cela n'est pas déjà fait)

- Télécharger l'application <https://www.scilab.org/download/6.0.2>.
Choisir l'installation suivant le système d'exploitation (Linux, Windows, OS,...)
- Utiliser Scilab en ligne <https://www.rollapp.com/app/scilab>.
Launch online puis ouvrir un compte (sign up) avec nom et mail.

1 Autour des nombres premiers

1.1 Conjecture de Goldbach

1.1.1 Petit historique et énoncé

Formulée en 1742 par Christian Goldbach, mathématicien prusse, la conjecture de Goldbach est l'un des plus vieux problèmes non résolus de la théorie des nombres et des mathématiques en général, tout comme l'hypothèse de Riemann et la conjecture des nombres premiers jumeaux ; ils sont référencés numéro 8 des problèmes que David Hilbert a énoncé en 1900.

Cette conjecture affirme que tout nombre pair supérieur ou égal 4 peut se décomposer comme la somme de deux nombres premiers.

Ce problème est extrêmement difficile, les mathématiciens essayent d'établir des résultats plus "faibles", par exemple que tout entier pair est somme d'au plus un nombre pair > 2 de nombres premiers ou que tout nombre impair > 5 est somme de d'un nombre impair de nombre premiers impairs.

Parmi les progrès récents sur la conjecture, signalons (de manière non exhaustive)

- en 1995, Olivier Ramaré, aujourd'hui en poste au CNRS à Marseille (il était précédemment à Lille) a démontré que tout entier pair est somme de six nombres premiers au plus. Corollaire : tout entier impair est somme de sept nombres premiers au plus.¹
- en 2013, Harald Helfgott (au CNRS à Paris) a démontré que tout entier impair > 5 est somme de trois nombres premiers. Corollaire : résultat en 2012 de Terence Tao (médaille Fields 2006).

1.1.2 Exercice

Le but de cet exercice n'est bien sûr pas de tenter démontrer ce résultat (auquel se heurtent encore des gens vraiment très très forts, vous l'aurez compris) mais de l'illustrer à l'aide de Scilab.

¹O. Ramaré est venu récemment deux fois à Amiens donner un exposé.

On utilisera bien entendu le crible d'Eratostène, il est fourni à la fin de ce texte. Nous proposons la méthode suivante : Soit i un entier fixé (pour le moment), on note $A = \{p \in \mathbb{N}, p \text{ premier}\}$.

Algorithm 1 : Conjecture de Goldbach pour i nombre pair fixé

```

1: for  $k = 2, \dots, i/2$  do
2:   if  $i - k = p \in A$  then
3:     le couple  $(k, p)$  convient
4:   end if
5: end for

```

Exercice 1 (Vérification de la conjecture de Goldbach jusqu'à un certain rang)

1. Construire un programme mettant en œuvre l'Algorithme 1 ; on inclura la fonction crible d'Eratostène pour déterminer les nombres premiers inférieurs à N . On utilisera la commande **find** fournie en annexe également.
2. Vérifier la conjecture de Goldbach pour quelques nombres pairs de votre choix.
3. Pour chaque entier pair, calculer le nombre de solutions. On programmera l'algorithme

Algorithm 2 : Conjecture de Goldbach pour tout entier pair inférieur à N

```

1: for  $i = 2, \dots, N$  par pas de deux do
2:   for  $k = 2, \dots, i/2$  do
3:     if  $i - k = p \in A$  then
4:       le couple  $(k, p)$  convient
5:       afficher le triplet  $(i, k, p)$ 
6:     end if
7:   end for
8: end for

```

4. (Facultatif) Représenter graphiquement le nombre de solutions trouvées pour chaque entier pair, on prendra $N = 100$.

1.2 Ordre de Grandeur des nombres premiers

On rappelle que $\pi(n)$ désigne le nombre de nombres premiers inférieurs ou égaux à n . On s'intéresse ici à la valeur du n -ième nombre premier noté p_n . On sait d'après le théorème des nombres premiers que $\pi(m) \simeq \frac{m}{\ln(m)}$, ainsi

$$p_{\pi(m)} \simeq \pi(m) \ln(\pi(m)).$$

On en déduit que $p_n \simeq n \ln(n)$ c'est à dire que $\lim_{n \rightarrow +\infty} \frac{p_n}{n \ln(n)} = 1$. On dispose des résultats plus précis suivants :

- $n \ln(n) < p_n < 1.7n \ln(n)$. Ce résultat, simple, est du à U. Felgner en 1990
- p_n est approché par $p'_n = n \ln(n) + n(\ln(\ln(n)) - 1)$. Ce résultat est du à J.B Rosser en 1938.

Exercice 2

1. A l'aide du crible d'Eratostène, calculer les 500 premiers nombres premiers (il faut aller jusqu'à 3600).
2. Ecrire un programme Scilab qui calcule $\pi(n)$ pour n variant de 1 à 500. Comparer graphiquement les ensembles de points $(i, \pi(i))$ et $(i, \frac{i}{\ln(i)})$, $i = 2, \dots, 500$. Conclusion ?
3. On illustre le résultat de Felgner. Représenter graphiquement l'encadrement de p_i par $i \ln(i)$ et $1.7i \ln(i)$. Conclusion ?
4. On illustre le résultat de Rosser.
 - (a) Pour i variant de 1 à 500, calculer $[p'_i], e_i = |p_i - [p'_i]|$ et $er_i = \frac{e_i}{p_i}$. Ici $[p'_i]$ désigne la partie entière de p'_i .
 - (b) Représenter graphiquement (i, e_i) et (i, er_i) pour $i = 1, \dots, 500$. Que constatez-vous ?

2 Approximations de constantes arithmétiques - quelques formules de Ramanujan

Voici quelques formules dues à Srinivasa Ramanujan (1887-1920).

- $\pi \simeq \frac{12}{\sqrt{130}} \ln \left(\frac{((3 + \sqrt{13})(\sqrt{8} + \sqrt{10}))}{2} \right)$ à 10^{-18} près
- $\pi \simeq \frac{9801\sqrt{2}}{4412}$ à 10^{-7} près
- $\frac{1}{\pi} = \frac{2\sqrt{2}}{9801} \sum_{n=0}^{+\infty} \frac{(4n)!}{(n!)^4} \times \frac{1103 + 26390n}{(4 \times 99)^{4n}}$

Exercice 3

Programmer ces formules et construire la suite. On utilisera le format `format("v",25)`.

3 Suites amusantes

3.1 Suite de Syracuse

3.1.1 Petit historique

La construction de cette suite est due à Lothar Collatz, elle se diffusa quand il la présenta à Helmut Hasse qui, à son tour, en fit part à ses collègues de l'Université de Syracuse aux Etats Unis, rien à voir donc avec la ville d'Archimède en Sicile. Par la suite d'éminents mathématiciens comme Stefan Ulam ou Paul Erdos s'y intéressèrent.

3.1.2 L'énoncé

Soit $N \in \mathbb{N}$. On définit la suite $(u_k)_{k \geq 0}$ comme suit :

$$u_0 = N, \text{ pour } k \geq 0 \quad u_{k+1} = \begin{cases} 3u_k + 1 & \text{si } u_k \text{ est impair,} \\ u_k/2 & \text{si } u_k \text{ est pair,} \end{cases}$$

La conjecture qui tient toujours, affirme que pour tout N donné, la suite retourne en 1 en temps fini, c'est à dire que, pour $u_0 = N$ donné il existe n tel que $u_n = 1$.

Exercice 4 (Suite de Syracuse)

1. Que se passe-t-il si $N = 1$?
2. Construire une fonction SYRAC en Scilab qui pour N donné retourne les termes de la suite de Syracuse jusqu'à rencontrer 1.
3. Après en avoir fait une copie, modifier la fonction SYRAC pour qu'elle retourne l'indice n du premier terme égal à 1.
4. Représenter le graphe (N, u_n) pour certaines valeurs de N entre 1 et 150.
5. Temps de vol. L'indice n est le temps de vol, c'est à dire le nombre d'itérations pour arriver à la valeur 1, bien entendu n dépend de N , mais on ne sait pas comment ! Représenter le graphe de (N, u_n) pour $N = 1, \dots, 150$.

3.2 Autres suites

Exercice 5

On rappelle que pour une suite convergente x_k donnée, on définit son facteur de convergence numérique par

$$\frac{x^{(k+1)} - x^{(k)}}{x^{(k)} - x^{(k-1)}}.$$

Représenter graphiquement les suites suivantes , ainsi que leur facteur de convergence

1. $u_{k+1} = \frac{1}{2} \left(u_k + \frac{2}{u_k} \right)$
2. $u_{k+1} = \sqrt{1 + u_k}$, puis $v_k = \frac{u_{k+1}}{u_k}$

Exercice 6

On considère la suite

$$u_k = \left(1 + \frac{1}{k} \right)^k.$$

1. Calculer numériquement les 100 premiers termes de la suite et la représenter numériquement.
2. Montrer que

$$u_n = e \left(1 - \frac{1}{2n} + \frac{11}{24n^2} - \frac{21}{48n^3} \right) + o\left(\frac{1}{n^3}\right).$$

3. Proposer une vérification numérique de cette relation.
4. Accélération. Trouver les réels a et b tels que

$$v_n = au_n + bu_{2n} = e + o\left(\frac{1}{n^2}\right)$$

et programmer la nouvelle suite v_n .

4 Annexe

4.1 Crible d'Eratostène

```
function A=erato (n)
A=2:n;// n entier >=2.
k=1;// k compte les nombres premiers trouvés.
while A(k)^2<=A($)
ms=max(size(A));
I=find(pmodulo(A(k+1 :ms) ,A(k))==0);
// On repere les multiples de A(k) après A(k).

A(k+I )=[]; // On a efface les multiples de A(k ).

k=k+1;
end ;
endfunction
```

4.2 Chercher un élément dans une liste

Soit V in vecteur. L'instruction $I=\text{find}(V)$ permet de trouver les indices i de V tels que $V(i) = 0$. Plus largement si $B(V)$ est un vecteur à valeurs booléennes (vrai ou faux), $\text{find}(B(V))$ retourne les indices tels que $B(V)$ est vrai. Exemple $V=[1, 4, 7, 9, -1]$ et $B(V) = V > 2$, $\text{find}(B(V))$ retournera les indices i tels que $V_i > 2$, c'est à dire $[2,3,4]$.