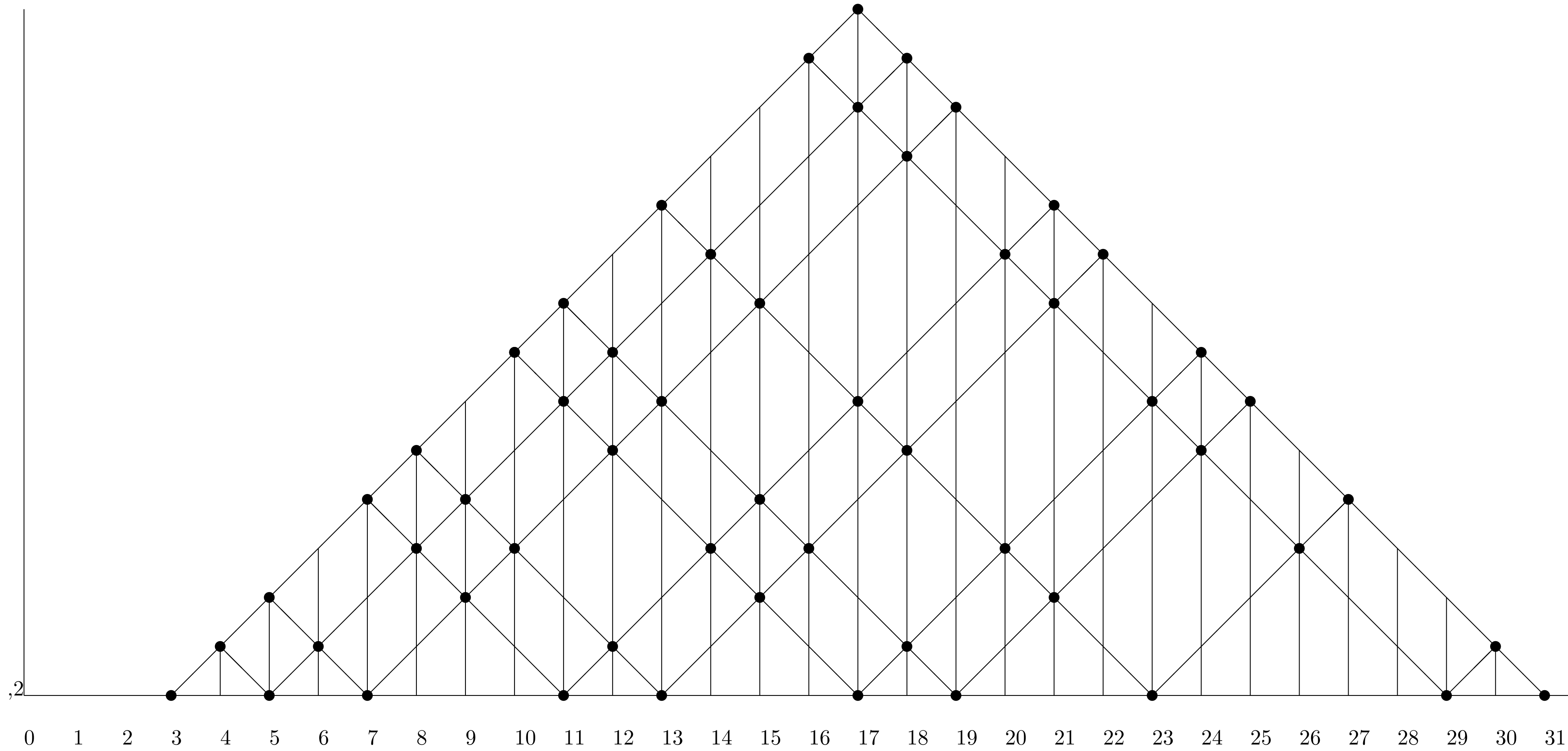


Figure 1: Le treillis Goldbach

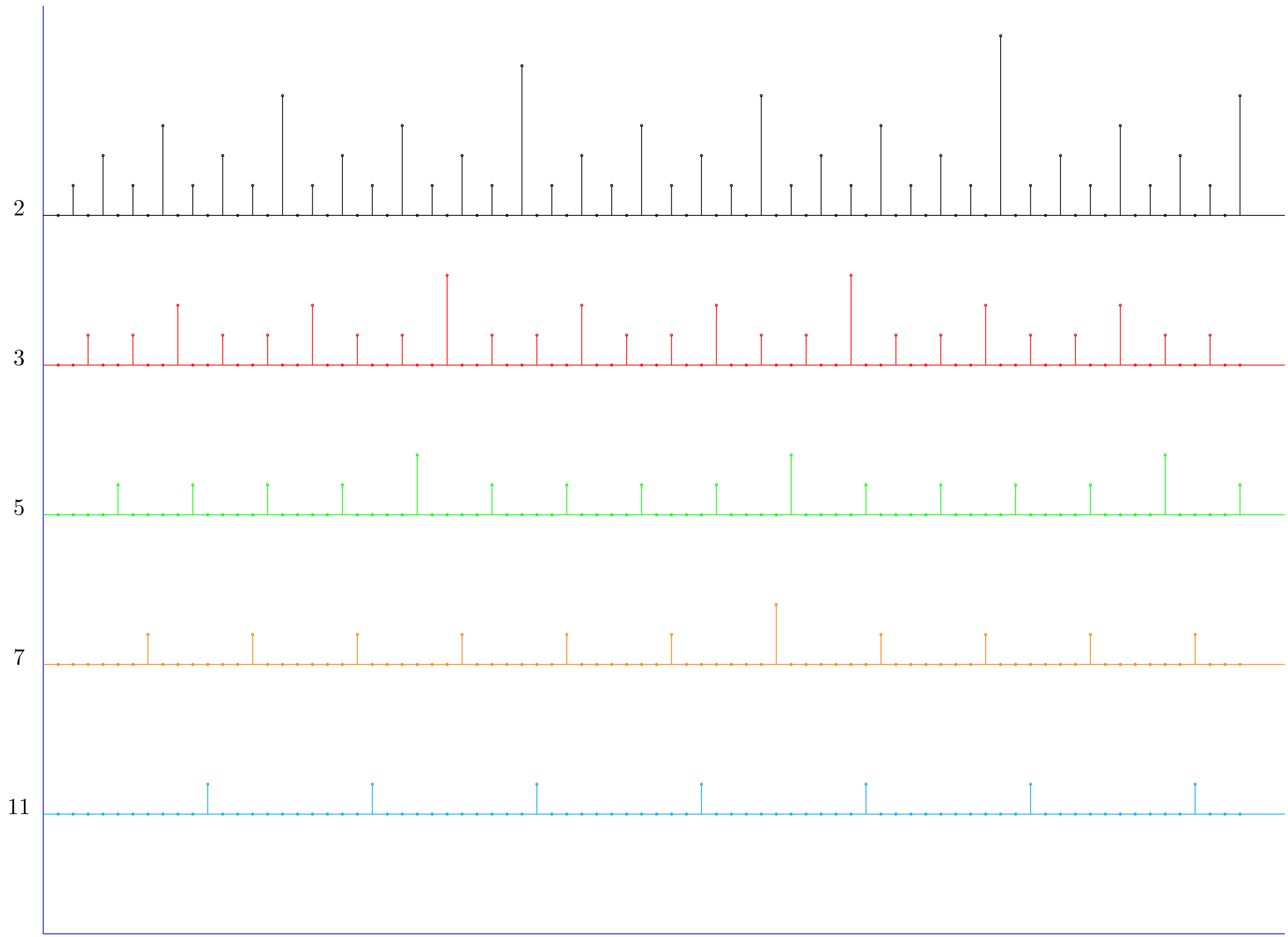
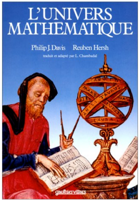


Figure 1 : Séquences fractales de valuations p-adiques

Le partage de *dg*

- $$\begin{array}{ll}
20902 = 3 + 20899 & 20962 = 3 + 20959 \\
20904 = 5 + 20899 & 20964 = 5 + 20959 \\
20906 = 3 + 20903 & 20966 = 3 + 20963 \\
20908 = 5 + 20903 & 20968 = 5 + 20963 \\
20910 = 7 + 20903 & 20970 = 7 + 20963 \\
20912 = 13 + 20899 & 20972 = 13 + 20959 \\
20914 = 11 + 20903 & 20974 = 11 + 20963 \\
20916 = 13 + 20903 & 20976 = 13 + 20963 \\
20918 = 19 + 20899 & 20978 = 19 + 20959 \\
20920 = 17 + 20903 & 20980 = 17 + 20963 \\
20922 = 19 + 20903 & 20982 = 19 + 20963 \\
20924 = 3 + 20921 & 20984 = 3 + 20981
\end{array}$$



- Des causes différentes produisent les mêmes effets (écart de 60, congrus mod 3 et 5).

Permutations des racines

- $$\begin{tikzcd} 2p_i \arrow[r, "f"] \arrow[d, "g_t"] & 2c \arrow[d, "g"] \\ p_i \arrow[r, "f"] & p_j \end{tikzcd}$$

- $$\begin{tikzcd} 94 = (1,4,3) \arrow[r, "f"] \arrow[d, "g_t"] & 88 = (1,3,4) \arrow[d, "g"] \\ 47 = (2,2,5) \arrow[r, "f"] & 29 = (2,4,1) \end{tikzcd}$$

$$\begin{tikzcd} \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z} \times \mathbb{Z}/7\mathbb{Z} \arrow[r, "f"] \arrow[d, "g_t"] & \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z} \times \mathbb{Z}/7\mathbb{Z} \arrow[d, "g"] \\ \mathbb{Z}/3\mathbb{Z} \setminus \{0,1\} \times \mathbb{Z}/5\mathbb{Z} \setminus \{0,4\} \times \mathbb{Z}/7\mathbb{Z} \setminus \{0,3\} \arrow[r, "f"] & \mathbb{Z}/3\mathbb{Z} \setminus \{0,1\} \times \mathbb{Z}/5\mathbb{Z} \setminus \{0,3\} \times \mathbb{Z}/7\mathbb{Z} \setminus \{0,4\} \end{tikzcd}$$

Permutations des racines

- $$\begin{tikzcd} 94 = (1,4,3) \arrow[r, "f"] \arrow[d, "g_t"] & 88 = (1,3,4) \arrow[d, "g"] \\ 47 = (2,2,5) \arrow[r, "f"] & 29 = (2,4,1) \end{tikzcd}$$

- $\mathbb{Z}/3\mathbb{Z} \rightarrow Id$,
  $\mathbb{Z}/5\mathbb{Z} \rightarrow \begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 0 & 1 & 4 & 2 & 3 \end{pmatrix}$,
  $\mathbb{Z}/7\mathbb{Z} \rightarrow \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 \\ 0 & 5 & 2 & 4 & 3 & 1 & 6 \end{pmatrix}$

Conclusion

- On a utilisé un SNURPF : un Système de NUmération par les Restes dans les Parties Finies de $\mathbb{N}$.

- On se situe dans une théorie lexicale des nombres, selon laquelle les nombres sont des mots.

Continuer de suivre Galois (DV - 21/12/2013)

On cherche une équation polynomiale qui aurait ses racines qui se verraient permutées par une certaine fonction et dont les solutions seraient les décomposants de Goldbach de n, un nombre pair, i.e. les nombres premiers dont les complémentaires à n seraient premiers également.

On “sent bien” que le générateur doit sûrement être la fonction $f : x \mapsto n - x$ car cette fonction envoie chaque nombre entier sur son complémentaire à n, la somme de ces deux nombres permettant d'obtenir n.

On trouve donc l'inéquation polynomiale $x^2 - nx \neq 0$ qui est invariante par la fonction f. En effet, $(n-x)^2 - n(n-x) = x^2 + n^2 - 2nx - n^2 + nx = x^2 - nx$. On est conforté dans cette idée par le fait que le polynôme proposé est égal à $x(n-x)$:

- d'une part, ce polynôme s'annule lorsque x est nul et la congruence $x \not\equiv 0 \ (mod\ p_i)$ dans tous les corps premiers $\mathbb{Z}/p_i\mathbb{Z}$ pour p_i un nombre premier quelconque inférieur à $\sqrt{n}$ correspond au fait que x est un nombre premier supérieur à $\sqrt{n}$;

- d'autre part, ce polynôme s'annule lorsque $x = n$ et la congruence $x \not\equiv n \ (mod\ p_i)$ dans tous les corps premiers $\mathbb{Z}/p_i\mathbb{Z}$ pour p_i un nombre premier quelconque inférieur à $\sqrt{n}$ correspond au fait que le complémentaire de x à n est premier.

Il faudrait pour prouver la conjecture de Goldbach être assuré que cette inéquation polynomiale $x^2 - nx \neq 0$ a une solution commune inférieure à $n/2$ dans tous les corps premiers $\mathbb{Z}/p_i\mathbb{Z}$ avec p_i un nombre premier quelconque inférieur à $\sqrt{n}$.

Traitons l'exemple de la recherche des décompositions de Goldbach de 98.
Le polynôme $x^2 - 98x$ est égal à $x^2 - 2x$ dans $\mathbb{Z}/3\mathbb{Z}$ tandis qu'il est égal à $x^2 - 3x$ dans $\mathbb{Z}/5\mathbb{Z}$, ou encore égal à x^2 tout simplement dans $\mathbb{Z}/7\mathbb{Z}$ puisque 7 divise 98.

Notons dans un tableau pour les nombres premiers supérieurs à $\sqrt{98}$ et inférieurs à 49 la moitié de 98 les valeurs des polynômes en question et voyons ceux qui sont éliminés dans chacun des corps premiers.

	11	13	17	19	23	29	31	37	41	43	47
x^2 (dont on teste la nullité dans $\mathbb{Z}/7\mathbb{Z}$)	121	169	289	361	529	841	961	1369	1681	1849	2209
$x^2 - 2x$ (dont on teste la nullité dans $\mathbb{Z}/3\mathbb{Z}$)	~~99~~	143	~~255~~	323	~~483~~	~~783~~	899	1295	~~1599~~	1763	~~2115~~
$x^2 - 3x$ (dont on teste la nullité dans $\mathbb{Z}/5\mathbb{Z}$)	88	~~130~~	238	304	~~460~~	754	868	1258	1558	~~1720~~	2068

On voit que ne sont conservés que les nombres 19, 31 et 37 qui sont comme attendu les décomposants de Goldbach de 98.

Le problème de Goldbach est en quelque sorte un problème “relatif” (puisqu'à la recherche des décomposants de Goldbach de n le nombre n intervient dans l'inéquation dont il faut chercher une solution commune dans tous les corps finis $\mathbb{Z}/p_k\mathbb{Z}$ pour $p_k \leqslant \sqrt{n}$).

On peut considérer que le problème des jumeaux est quant à lui le problème “absolu” correspondant au problème “relatif” de Goldbach. En effet, si l'on appelle “père de jumeaux” le nombre pair entre deux nombres premiers jumeaux (par exemple 18 entre 17 et 19 ou encore 570 entre 569 et 571), ce nombre doit vérifier l'inéquation “absolue” $x^2 \not\equiv 1 (mod\, p_k)$ pour tout $p_k \leqslant \sqrt{x+1}$ (il doit en effet vérifier simplement $(x-1)(x+1) \not\equiv 0 (mod\, p_k)$ pour qu'$x - 1$ et $x + 1$ soient premiers tous les deux). Un père de jumeau est obligatoirement de la forme $6k$. Fournissons dans un tableau la classe de congruence de x^2 selon les modules premiers impairs inférieurs à $\sqrt{x+1}$ qui nous permettent d'aisément trouver les pères de jumeaux jusqu'à 300.

père	*mod* 3	*mod* 5	*mod* 7	*mod* 11	*mod* 13	*mod* 17	*jumeaux*
6							$(5, 7)$
12	0						$(11, 13)$
18	0						$(17, 19)$
24	0	1					
30	0	0					$(29, 31)$
36	0	1					
42	0	4					$(41, 43)$
48	0	4	1				
54	0	1	4				
60	0	0	2				$(59, 61)$
66	0	1	2				
72	0	4	4				$(71, 73)$
78	0	4	1				
84	0	1	0				
90	0	0	1				
96	0	1	4				
102	0	4	2				$(101, 103)$
108	0	4	2				$(107, 109)$
114	0	1	4				
120	0	0	1	1			
126	0	1	0	3			
132	0	4	1	0			
138	0	4	4	3			$(137, 139)$
144	0	1	2	1			
150	0	0	2	5			$(149, 151)$
156	0	1	4	4			
162	0	4	1	9			
168	0	4	0	9	1		
174	0	1	1	4	12		
180	0	0	4	5	4		$(179, 181)$
186	0	1	2	1	3		
192	0	4	2	3	9		$(191, 193)$
198	0	4	4	0	9		$(197, 199)$
204	0	1	1	3	3		
210	0	0	0	1	4		
216	0	1	1	5	12		
222	0	4	4	4	1		
228	0	4	2	9	10		$(227, 229)$
234	0	1	2	9	0		
240	0	0	4	4	10		$(239, 241)$
246	0	1	1	5	1		
252	0	4	0	1	12		
258	0	4	1	3	4		
264	0	1	4	0	3		
270	0	0	2	3	9		$(269, 271)$
276	0	1	2	1	9		
282	0	4	4	5	3		$(281, 283)$
288	0	4	1	4	4		
294	0	1	0	9	12	8	
300	0	0	1	9	1	2	

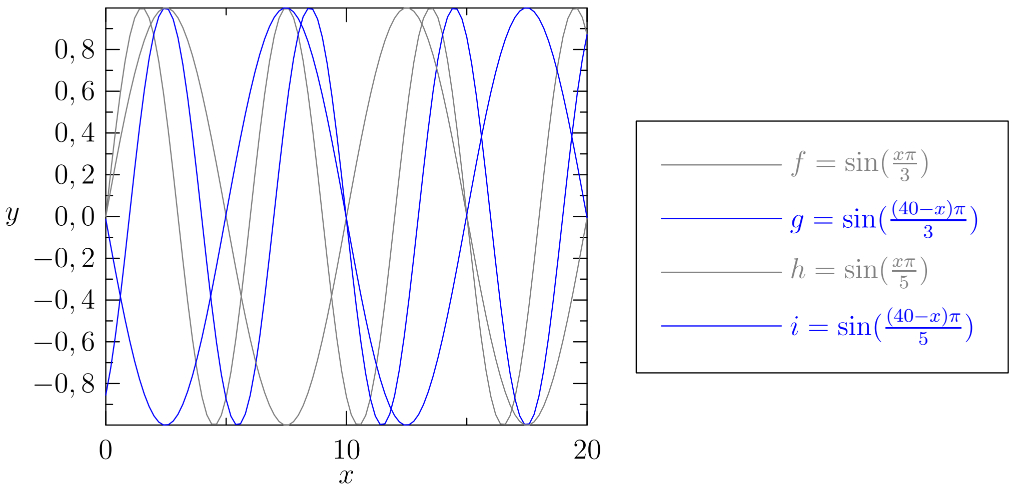
On ne réexplique pas ici la représentation par les grilles de divisibilité qui nous a permis de mieux comprendre la conjecture de Goldbach et dont l'exemple du nombre pair 40 est représenté ci-dessus. 11 et 17, qui ne sont divisibles ni par 3 ni par 5 et qui ne partagent avec 40 aucun de leur reste dans des divisions euclidiennes par 3 ou 5 (i.e. dont la colonne ne contient pas de case colorée) sont des décomposants de Goldbach de 40.

On a proposé à partir de ces grilles la possibilité de trouver les décomposants de Goldbach en calculant des produits de sinusoïdes.

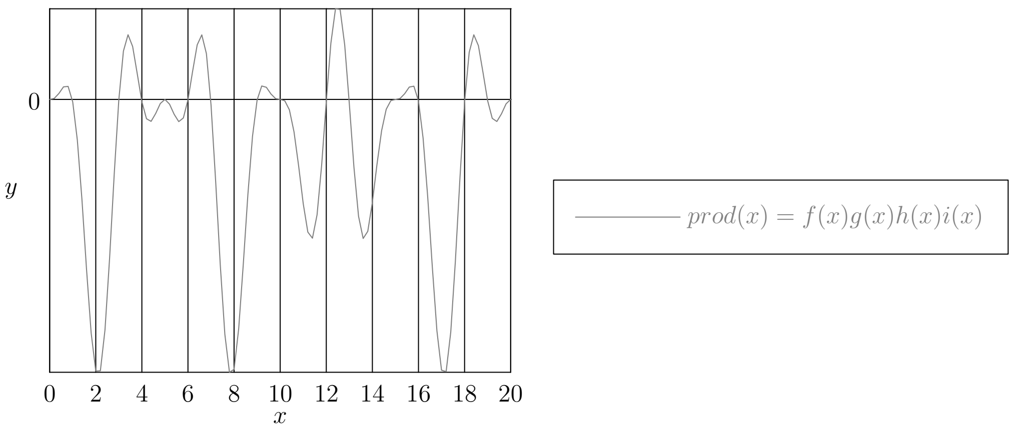
Les décomposants de Goldbach de n sont en effet les seuls nombres entiers impairs inférieurs à $n/2$ qui n'annulent pas le produit suivant :

$$\prod_{3 \leqslant p\ un\ nb\ 1^{er} \leqslant \sqrt{n}} sin\left(\frac{x\pi}{p}\right).sin\left(\frac{(n-x)\pi}{p}\right)$$

Les sinusoïdes correspondant au cas du nombre pair 40 (se reporter à la grille de divisibilité ci-dessus) sont :



Leur produit ne s'annule effectivement pas pour les nombres entiers impairs 11 et 17.

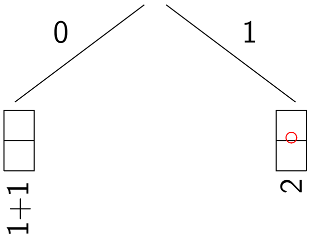


On peut établir une analogie entre ces sinusoïdes et les fonctions d'onde de la mécanique quantique. En poussant l'analogie, on peut imaginer qu'on puisse établir une probabilité qu'un nombre pair ait un décomposant de Goldbach sans pouvoir établir sa valeur, selon une sorte de principe d'incertitude.

Enfin, si on souhaite établir une analogie avec la propriété d'*intrication quantique* : on associe à chaque case de la grille de divisibilité ci-dessus un q-bit qui est simultanément dans les états 0 et 1. On peut imaginer cette grille comme de taille infinie si on considère tous les nombres pairs d'un même coup. Le fait de fixer

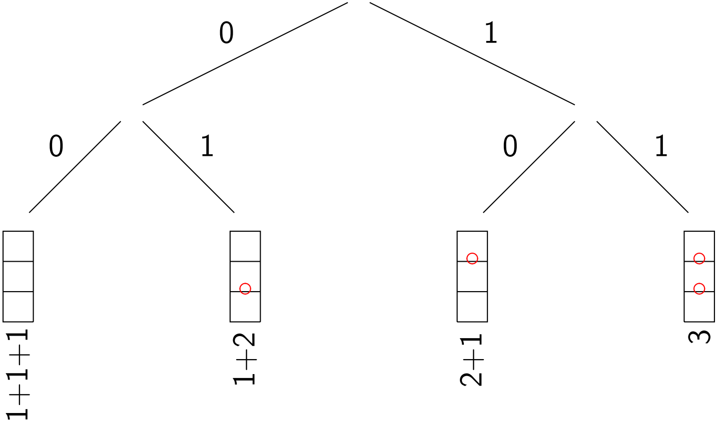
Compositions et mots booléens

- A chaque entier n, on associe ses 2^{n-1} compositions additives.
- *exemple* : arbre binaire des compositions de 2



Exemple

- *exemple* : arbre binaire des compositions de 3



- Noter que la composition $1 + 2$ est différente de la composition $2 + 1$.

Obtention des compositions de $n+1$ à partir de celles de n

- On concatène 0 ou 1 au début de chaque mot booléen de n.

- Cela correspond à deux actions syntaxiques : concaténer "1+" en début de composition ou bien remplacer le premier sommant par son successeur.

Nombre composé / nombre premier

- On appelle compositions triviales la composition correspondant au mot booléen ne contenant que des 0 (composition de la forme $1 + 1 + \ldots + 1$) ou bien la composition correspondant au mot booléen contenant $n - 1$ lettres 1 (composition de la forme n).

- Un nombre composé admet au moins une décomposition non triviale de la forme $x + x + x + \ldots + x$ contenant 2 occurrences de x au moins.

- A chaque entier est associé un ensemble de mots booléens, i.e. un ensemble de parties de $\mathbb{N}$.

$$\begin{aligned} \mathbb{N} &\longrightarrow \{0,1\}^{\{0,1\}^{\mathbb{N}}} \\ n &\longmapsto \{s \in \{0,1\}^{\mathbb{N}} / \forall i \geqslant n, s[i] = 0\} \subset \mathcal{P}(\mathbb{N}) \end{aligned}$$

Formalisation

- Le passage de n à $n+1$ est codé par le diagramme suivant :

$$\begin{tikzcd}
\mathbb{N} \arrow[r, "d_n"] \arrow[d, "+1"'] & \{0,1\}^{\{0,1\}^{\mathbb{N}}} \arrow[d, "d_{n+1}"] \\
\mathbb{N} \arrow[r, "d_n"'] & \{0,1\}^{\{0,1\}^{\mathbb{N}}}
\end{tikzcd}$$

avec $\qquad d_n : \mathbb{N} \longrightarrow \{0,1\}$

et

$$d_{n+1} : \begin{array}{ccc} k & \longmapsto & d_n(k-1), \forall k \geqslant 1 \\ 0 & \longmapsto & 0 \end{array}$$

- faire l'union de cet ensemble de fonctions avec l'ensemble des fonctions d_{n+1} qui associent l'image 1 (plutôt que 0) à 0.

Nombre composé / nombre premier

- Un nombre est composé n si l'un de ces mots non triviaux (dont on considère les $n-1$ premières lettres, i.e. la partie des mots avant l'infinité de zéros) admet une période (c'est un motif qui se répète, en théorie des langages).

- Un nombre est premier si tous ses mots sont apériodiques.

Pourquoi la moyenne des parties fractionnaires des parties réelles des zéros de zêta semble-t-elle être asymptotiquement égale à 1/2 ? (Denise Vella-Chemla, 7.3.2018)

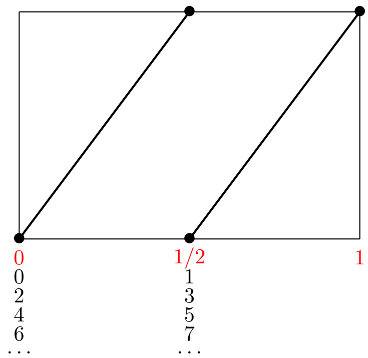
On voudrait fournir ici une explication informelle à une constatation effectuée récemment et qui est que la moyenne des parties fractionnaires des parties réelles des zéros de la fonction zêta semble tendre vers 1/2.

On utilise pour illustrer cette explication la notion de fonction ou signal en dents-de-scie dont on peut trouver une présentation dans les pages wikipedia en français (https://fr.wikipedia.org/wiki/Signal$_{e}n_{d}ents_{d}e_{s}cie$) ou en anglais (https://en.wikipedia.org/wiki/Sawtooth$_{w}ave$).

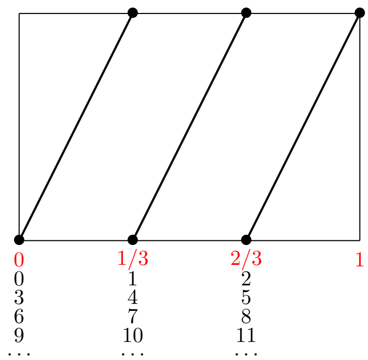
On pense au Snurpf, système de Numération par les Restes (modulaires de Gauss) dans les Parties Finies de $\mathbb{N}$. On conçoit aisément que les restes des divisions euclidiennes des nombres entiers successifs par 2 (en commençant par l'entier 1) sont 1, 0, 1, 0, 1, etc. De même les restes des divisions euclidiennes des entiers successifs par 3 sont 1, 2, 0, 1, 2, 0, 1, 2, 0, etc.

Pour représenter ces *fonctions croissantes "restes de divisions euclidiennes"* ainsi décrites (et qui retombent régulièrement à 0) , on se place sur l'intervalle $[0, 1]$ et on utilise des fonctions en dents-de-scie.

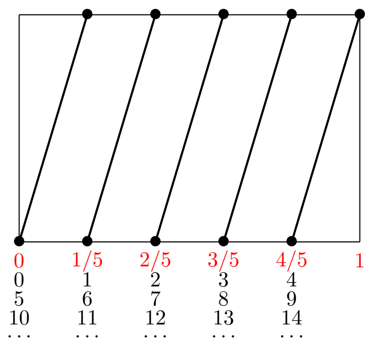
Voici la fonction en dents-de-scie qui permet de visualiser la notion de parité. Les entiers impairs ont pour image le point médian, les pairs ont pour image 0.



Voici la fonction en dents-de-scie qui permet de visualiser la notion de divisibilité par 3. Les entiers qui ont pour reste 1 dans la division euclidienne par 3 ont pour image le point 1/3, ceux qui ont pour reste 2 ont pour image le point 2/3, et les nombres divisibles par 3 ont pour image 0.



Voici la fonction en dents-de-scie qui permet de visualiser la notion de divisibilité par 5. Les nombres qui ont pour reste r dans la division euclidienne par 5 ont pour image $r/5$.



Comme les fonctions en dents-de-scie sont périodiques, il faut les imaginer sur un tore (ou du moins un cylindre). Sur les bords verticaux des représentations planes du cylindre ci-dessus vont se retrouver tous les nombres multiples d'un nombre donné, ainsi que le nombre en question. La périodicité qui fait que la fonction attribue la même image à deux nombres dont la différence vaut 1 s'exprime par la condition $x(t) = t \ (mod\ 1)$.

Il faut imaginer les nombres qui ne sont pas divisibles (par 3 par exemple) comme positionnés sur les lignes verticales de part et d'autre de la droite $x = 1/2$. Les nombres premiers doivent être équitablement répartis entre ces deux ordonnées de localisation (les densités associés à de tels sous-ensembles de nombres dans $\mathbb{N}$ doivent être égales). On imagine qu'on peut trouver une fonction qui envoie les entiers qui ont pour reste 1 dans une division euclidienne par 3 sur le "lieu" $x = 1/3$ et qui envoie les entiers qui ont pour reste 2 dans une division euclidienne par 3 sur le "lieu" $x = 2/3$.

On peut "mélanger" ou "agréger" toutes les fonctions en dents-de-scie en une seule fonction par la fonction inverse de la décomposition en série de Fourier. Du fait de l'équilibre des restes des divisions euclidiennes, les nombres doivent se répartir équitablement de part et d'autre de la droite $x = 1/2$. L'article wikipedia en anglais fournit comme renseignement supplémentaire : *"a sawtooth wave's sound is harsh and clear and its spectrum contains both even and odd harmonics of the fundamental frequency. Because it contains all the integer harmonics, it is one of the best waveforms to use for subtractive synthesis of musical sounds, particularly bowed string instruments like violins and cellos, since the slip-stick behavior of the bow drives the strings with a sawtooth-like motion."*.

Moyenne des parties fractionnaires des parties réelles des zéros de zêta (Denise Vella-Chemla, 4.3.2018)

On calcule par le programme python suivant les moyennes des parties fractionnaires des parties réelles des zéros de zêta qui sont fournies par Odlyzko ici http://www.dtc.umn.edu/ odlyzko/zeta$_{t}$*ables/index.html*.

```
import math
from math import *

zeros=[]
with open('leszerosdezeta', 'r') as f:
    for p in f.readlines():
        z = float(p.split()[1])
        zeros.append(z)
f.close()
print('')
somme=0.0
for i in range(0,100000):
    res = zeros[i]-floor(zeros[i])
    somme=somme+res
    print(res)
print('moyenne')
print(somme/i)
```

Pour le fichier contenant 100000 zéros, la moyenne obtenue est : 0.499356115471.

Pour le fichier des zéros autour de 10^{12}, la moyenne obtenue est : 0.499758301944.

Pour le fichier des zéros autour de 10^{21}, la moyenne obtenue est : 0.496515278444.

Pour le fichier des zéros autour de 10^{22}, la moyenne obtenue est : 0.4983445377.

Pour le très gros fichier fourni par Odlyzko et contenant 2001053 zéros de zêta, la moyenne obtenue est : 0.500277516477.

On refait quelques tests, en les complétant de tests sur un fichier de nombres aléatoires de l'intervalle $[0, 1]$.

On obtient pour le fichier contenant les 2001053 zéros fournis par Odlyzko :
- moyenne = 0.50027726647,
- médiane = 0.499993778489,
- écart-type = 0.288607100069.

On obtient pour le fichier contenant des nombres aléatoires :
- moyenne = 0.500009450015,
- médiane = 0.499661660179,
- écart-type = 0.28860413121.

```
import math
from math import *
import numpy
from numpy import *
import numpy.random

tabzeros=fromfile('leszerostresgrosfichier',dtype=float,count=-1,sep=' ')
tab2=numpy.random.random(tabzeros.size)
print('')
print(tabzeros.ndim)
print(tabzeros.size)
for i in range(tabzeros.size):
    tabzeros[i]=tabzeros[i]-floor(tabzeros[i])
print('tab1')
print(mean(tabzeros))
print(median(tabzeros))
print(std(tabzeros))
print('tab2')
print(mean(tab2))
print(median(tab2))
print(std(tab2))
```

$$exp(li(\kappa.t.i.o(n))).exp(e^{\partial i.t.i^{\nu e}})$$

Les zéros de zêta ne peuvent pas être ailleurs que sur la droite critique parce que sinon, admettons qu'il y en ait à peine 2 qui ne soient pas sur la droite des nombres de partie réelle $\frac{1}{2}$, que l'un soit de la forme $\rho = \alpha + \beta i$ et que son conjugué soit de la forme $\alpha - \beta i$, quand, dans la formule de Riemann

$$f(x) = Li(x) - \sum_{\rho} (Li(x^{\rho}) + Li(x^{\overline{\rho}})) + \int_x^{\infty} \frac{du}{u(u^2-1)ln\, u} - ln\, 2,$$

on ajouterait les logarithmes intégrals de ces 2 zéros conjugués, alors que lorsque les zéros appartiennent bien à la droite critique, les parties imaginaires s'annulent entraînant l'ajout seulement du double de la partie réelle commune aux deux zéros, là, pour ces deux zéros hors droite critique, on obtiendrait comme valeur de la somme des logarithmes intégrals de ces deux complexes un nombre complexe, et alors on serait obligé de conclure par une phrase du style "jusqu'à 3 000 456 278, il y a $5\ 678 + 8\ 528i$ nombres premiers, ce qui, on l'avouera, ne veut strictement rien dire". Nous ne voyons pas d'autre explication...

Primalité et zéros de sommes de cosinus

Denise Vella-Chemla

10/7/14

L'article d'Euler *Découverte d'une loi tout extraordinaire des nombres par rapport à la somme de leurs diviseurs* est magique. On reste subjugué par la manière dont le mathématicien a trouvé la formule récurrente de la somme des diviseurs.

On peut trouver sur un forum de mathématiques [1] une autre formule :

$$\sigma(n) = \sum_{k=1}^{n} \sum_{l=1}^{k} cos\left(\frac{2\pi nl}{k}\right)$$

Les cosinus se comportent ici comme des booléens qui comptent pour chaque diviseur sa valeur comme une somme de 1, de façon analogue à la fonction *Succ* de l'arithmétique de Peano.

Du coup, on en déduit une manière rigolote de trouver les nombres premiers : ce sont les zéros de la fonction *sumsumcos* ci-dessous :

$$sumsumcos(n) = \sum_{k=2}^{n-1} \sum_{l=1}^{k} cos\left(\frac{2\pi nl}{k}\right)$$

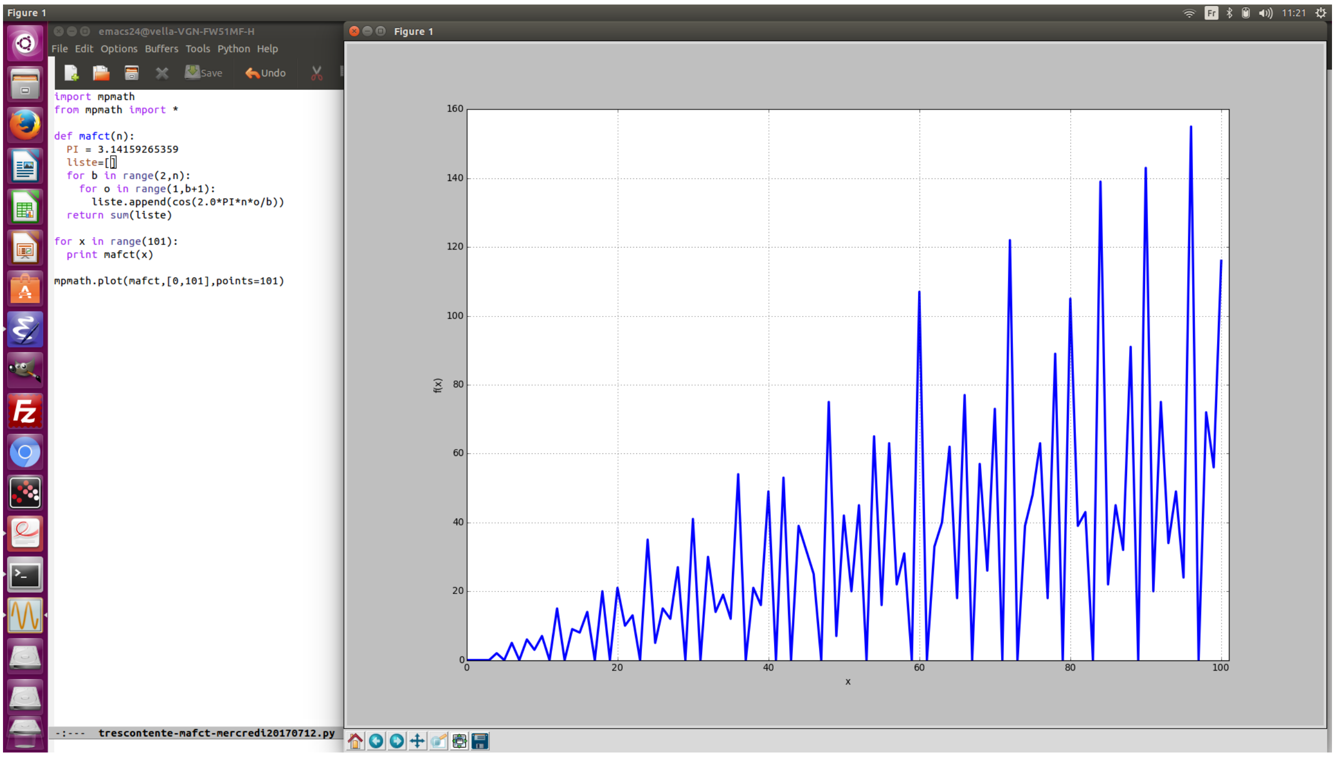
For me, that's F_{un} !

1. à l'adresse $http://www.les-mathematiques.net/phorum/read.php?5,892412,892412$.

Alterner les termes de la somme de cosinus qui s'annule pour les nombres premiers (Denise Vella-Chemla, 28.10.2018)

On a proposé en juillet 2014 * la caractérisation suivante des nombres premiers (motivée par le fait que si p est un nombre premier, $\sigma(p) = p + 1$ †) :

$$n \text{ } est \text{ } premier \iff \sum_{b=2}^{n-1} \sum_{o=1}^{b} cos \frac{2\pi no}{b} = 0$$



On trouve ici une démonstration du fait que cette caractérisation des nombres premiers en est bien une : http://denise.vella.chemla.free.fr/VictorVarinKeldyshSumsumcos.pdf.

Il s'agit d'une caractérisation triviale des nombres premiers dans le sens où elle calcule la somme des diviseurs des entiers successifs par le biais de calcul d'angles et par le test du fait que ces angles sont ou non multiples de 2π. Cela revient au même pour tester si un nombre est premier qu'à étudier sa division par tous les entiers qui lui sont inférieurs ‡.

Voyons comment la somme de cosinus calcule la somme des diviseurs de 2, 3, 4 et 5. On note les angles en degrés pour que la divisibilité se voie mieux. θ dénote les angles dont sont calculés les cosinus.

*. cf http://denise.vella.chemla.free.fr/primesommecos.pdf.
†. $\sigma(n)$ est la notation habituelle pour la somme des diviseurs de n.
‡. Cf page 9 de la première note qu'on avait écrite au sujet de la conjecture de Goldbach en septembre 2005.

n	$o, b \to \theta$	cos	$o, b \to \theta$	cos	$o, b \to \theta$	cos	$o, b \to \theta$	cos	$o, b \to \theta$	cos	$\sigma(n)$
2	$1, 1 \to 720$	1									
	$1, 2 \to 360$	1	$2, 2 \to 720$	1							3
3	$1, 1 \to 1080$	1									
	$1, 2 \to 540$	-1	$2, 2 \to 1080$	1							
	$1, 3 \to 360$	1	$2, 3 \to 720$	1	$3, 3 \to 1080$	1					4
4	$1, 1 \to 1440$	1									
	$1, 2 \to 720$	1	$2, 2 \to 1440$	1							
	$1, 3 \to 480$	-0.5	$2, 3 \to 960$	-0.5	$3, 3 \to 1440$	1					
	$1, 4 \to 360$	1	$2, 4 \to 720$	1	$3, 4 \to 1080$	1	$4, 4 \to 1440$	1			7
5	$1, 1 \to 1800$	1									
	$1, 2 \to 900$	-1	$2, 2 \to 1800$	1							
	$1, 3 \to 600$	-0.5	$2, 3 \to 1200$	-0.5	$3, 3 \to 1800$	1					
	$1, 4 \to 450$	0	$2, 4 \to 900$	-1	$3, 4 \to 1350$	0	$4, 4 \to 1800$	1			
	$1, 5 \to 360$	1	$2, 5 \to 720$	1	$3, 5 \to 1080$	1	$4, 5 \to 1440$	1	$5, 5 \to 1800$	1	6

On constate par programme qu'en alternant les signes $+$ et $-$ devant chaque terme de la somme et en soustrayant 1 au résultat global, on obtient que les nombres premiers de la forme $4k+1$ ont pour image 0 quand les nombres premiers de la forme $4k+3$ ont pour image 1 selon le programme suivant :

```
import mpmath
from mpmath import *

def mafct(n):
  oppose = 1
  liste=[]
  for b in range(2,n):
    for o in range(1,b+1):
      oppose = (-1) * oppose
      liste.append(oppose*cos(2*pi*n*o/b))
  res=sum(liste)-1
  return res

for x in range(101):
  print(str(x)+' a pour somme '+str(mafct(x)))

mpmath.plot(mafct,[0,101],points=101)
```

Résultat du programme ci-dessus : calcul d'une somme alternée de cosinus

```
1 a pour somme -1
2 a pour somme -1
3 a pour somme 1.0
4 a pour somme -2.0
5 a pour somme -2.66453525910038e-15
6 a pour somme -5.0
7 a pour somme 0.999999999999997
8 a pour somme -2.0
9 a pour somme 6.0
10 a pour somme -5.00000000000001
```

```
11 a pour somme 1.00000000000002
12 a pour somme -10.0
13 a pour somme 1.86517468137026e-14
14 a pour somme -4.99999999999998
15 a pour somme 16.9999999999999
16 a pour somme -2.00000000000003
17 a pour somme 1.06581410364015e-14
18 a pour somme -17.0
19 a pour somme 1.00000000000006
20 a pour somme -10.0000000000001
21 a pour somme 20.0
22 a pour somme -5.00000000000012
23 a pour somme 1.00000000000001
24 a pour somme -18.0000000000001
25 a pour somme 9.99999999999995
26 a pour somme -4.99999999999988
27 a pour somme 24.9999999999999
28 a pour somme -10.0000000000001
29 a pour somme -8.21565038222616e-14
30 a pour somme -36.9999999999999
31 a pour somme 1.00000000000026
32 a pour somme -2.00000000000002
33 a pour somme 27.9999999999999
34 a pour somme -4.99999999999991
35 a pour somme 25.0
36 a pour somme -33.9999999999997
37 a pour somme -3.19744231092045e-14
38 a pour somme -5.00000000000025
39 a pour somme 33.0000000000003
40 a pour somme -17.9999999999999
41 a pour somme 1.75859327100625e-13
42 a pour somme -45.0
43 a pour somme 1.0000000000002
44 a pour somme -9.9999999999999
45 a pour somme 64.0000000000001
46 a pour somme -5.00000000000072
47 a pour somme 1.00000000000015
48 a pour somme -34.0000000000001
49 a pour somme 14.0000000000003
50 a pour somme -24.9999999999995
51 a pour somme 40.9999999999997
52 a pour somme -9.99999999999957
53 a pour somme 1.93622895494627e-13
54 a pour somme -52.9999999999998
55 a pour somme 33.0000000000002
56 a pour somme -17.9999999999994
57 a pour somme 43.9999999999999
58 a pour somme -4.99999999999929
59 a pour somme 0.999999999999813
60 a pour somme -73.9999999999995
61 a pour somme -5.10924635932497e-13
62 a pour somme -4.99999999999921
63 a pour somme 81.0
64 a pour somme -2.00000000000007
65 a pour somme 35.999999999999
66 a pour somme -60.9999999999997
67 a pour somme 1.00000000000103
68 a pour somme -10.0000000000005
69 a pour somme 51.9999999999997
70 a pour somme -53.0000000000017
```

```
71 a pour somme 0.999999999999956
72 a pour somme -65.9999999999991
73 a pour somme 5.77315972805081e-13
74 a pour somme -4.99999999999963
75 a pour somme 97.0
76 a pour somme -10.0000000000007
77 a pour somme 36.0000000000004
78 a pour somme -68.9999999999998
79 a pour somme 1.00000000000032
80 a pour somme -33.9999999999996
81 a pour somme 78.0000000000006
82 a pour somme -5.00000000000226
83 a pour somme 0.999999999998838
84 a pour somme -89.9999999999989
85 a pour somme 43.9999999999991
86 a pour somme -4.99999999999942
87 a pour somme 64.9999999999984
88 a pour somme -18.0000000000019
89 a pour somme -5.15143483426073e-14
90 a pour somme -132.999999999998
91 a pour somme 41.0000000000008
92 a pour somme -9.99999999999778
93 a pour somme 67.9999999999987
94 a pour somme -5.00000000000054
95 a pour somme 48.9999999999985
96 a pour somme -65.9999999999998
97 a pour somme 1.24611432283928e-12
98 a pour somme -32.9999999999993
99 a pour somme 112.999999999998
100 a pour somme -50.0000000000001
```

Si l'on initialise le signe du premier terme de la somme à -1 plutôt qu'à $+1$ et si l'on ajoute 2 à la somme globale plutôt que de lui soustraire 1, alors les rôles des nombres premiers de la forme $4k+1$ et $4k+3$ sont échangés, les premiers ayant alors pour image 1 au lieu de 0 et les seconds ayant pour image 0 au lieu de 1 selon le programme, les graphiques et le tableau des images par les deux fonctions sommes alternées de cosinus suivants :

```
import mpmath
from mpmath import *

def mafct(n):
  oppose = -1
  liste=[]
  for b in range(2,n):
    for o in range(1,b+1):
      oppose = (-1) * oppose
      liste.append(oppose*cos(2*pi*n*o/b))
  res=sum(liste)-1
  return res

for x in range(101):
  print(str(x)+' a pour somme '+str(mafct(x)))

mpmath.plot(mafct,[0,101],points=101)
```

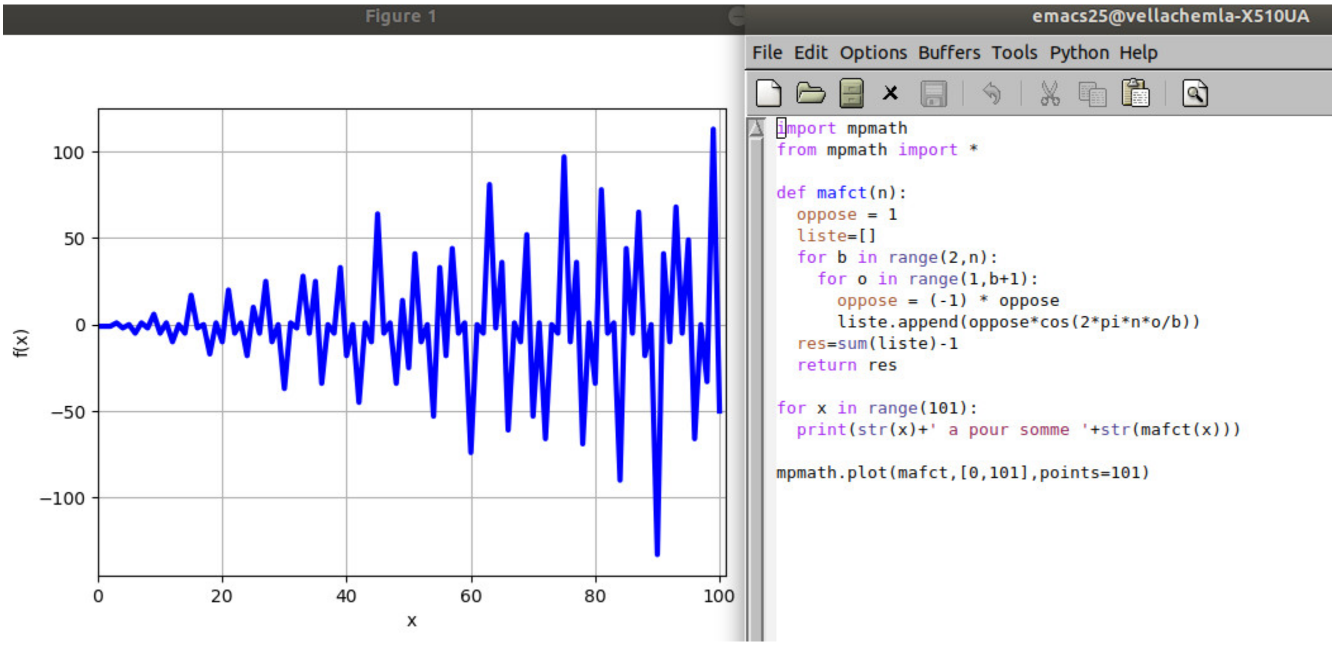


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
100
50
0
−50
−100
f(x)
0
20
40
60
80
100
x
import mpmath
from mpmath import *
def mafct(n):
oppose = 1
liste=[]
for b in range(2,n):
for o in range(1,b+1):
oppose = (-1) * oppose
liste.append(oppose*cos(2*pi*n*o/b))
res=sum(liste)-1
return res
for x in range(101):
print(str(x)+' a pour somme '+str(mafct(x)))
mpmath.plot(mafct,[0,101],points=101)

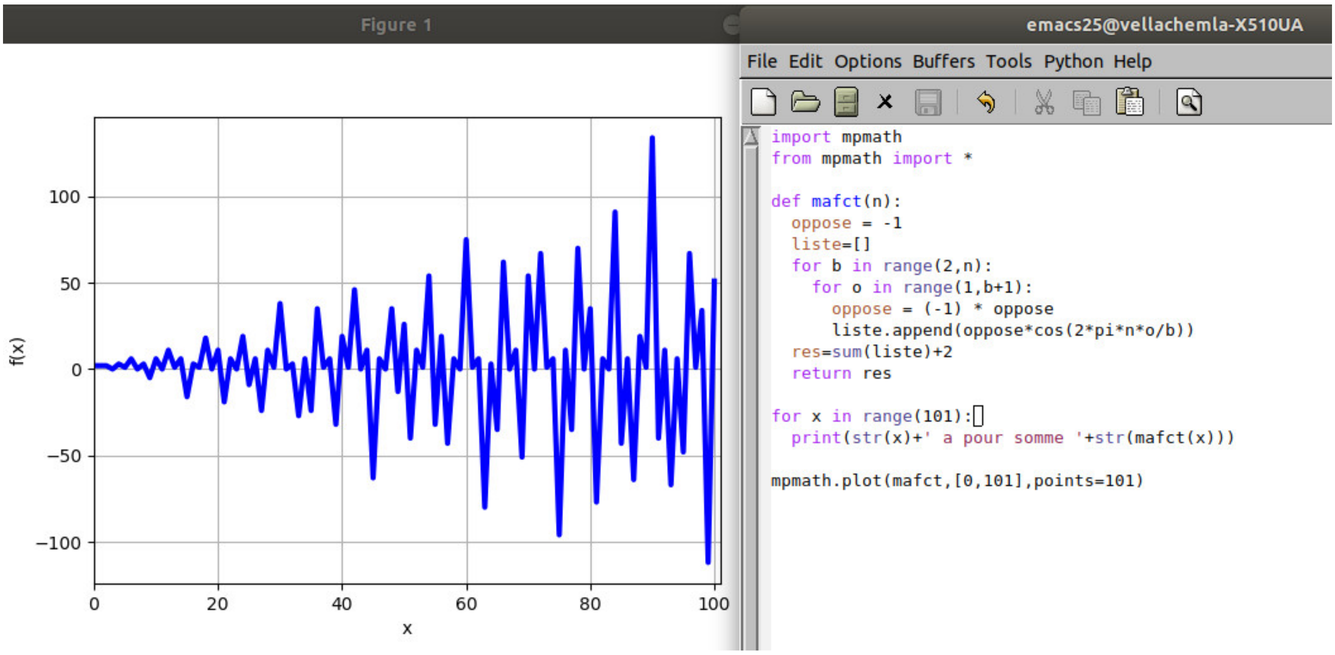


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
100
50
0
−50
−100
f(x)
0
20
40
60
80
100
x
import mpmath
from mpmath import *
def mafct(n):
oppose = -1
liste=[]
for b in range(2,n):
for o in range(1,b+1):
oppose = (-1) * oppose
liste.append(oppose*cos(2*pi*n*o/b))
res=sum(liste)+2
return res
for x in range(101):
print(str(x)+' a pour somme '+str(mafct(x)))
mpmath.plot(mafct,[0,101],points=101)

p	*somme alternée* 1	*somme alternée* 2
3	1	0
5	0	1
7	1	0
11	1	0
13	0	1
17	0	1
19	1	0
23	1	0
29	0	1
31	1	0
37	0	1
41	0	1
43	1	0
47	1	0
53	0	1
59	1	0
61	0	1
67	1	0
71	1	0
73	0	1
79	1	0
83	1	0
89	0	1
97	0	1

Annexe : extrait de la première note de septembre 2005 (il faut plutôt prendre le complexe 1 comme sommet commun des polygones)

Au tout début, nous réfléchissions à une manière élégante d'implémenter les horloges modulaires Gaussiennes. On peut voir l'horloge modulaire de n comme un polygone régulier à n côtés sur le cercle unité. Prenons comme convention que tous les polygones ont en commun le sommet correspondant à midi. Deux nombres sont premiers entre eux si leurs polygones réguliers respectifs n'ont aucun sommet commun hormis le sommet midi. Cette idée des polygones réguliers nous a fait faire un détour par les fractions à coefficients entiers. 4 n'est pas premier car $2/4 = 1/2$. Cela nous a amenée naturellement a nous rendre compte qu'un nombre était premier si toutes les fractions de $1/n$ à $(n-1)/n$ étaient non réductibles.

La considération des fractions entières $1/5$, $2/5$, $3/5$, $4/5$, nous a fait dériver vers les sinusoïdes. En effet, les sinusoïdes sont des fonctions qui passent régulièrement par zéro. La sinusoïde $\sin(5\pi x)$ s'annule justement pour les 4 fractions qui nous intéressent sur l'intervalle $]0,1[$. Un nombre n est ainsi premier si sa sinusoïde s'annule exactement $n-1$ fois dans l'intervalle $]0,1[$ et ce, jamais sur un point pour lequel s'annule la sinusoïde d'un nombre premier inférieur à lui.

Nous avons vite abandonné cette voie de recherche : le fait d'assimiler un nombre premier p à sa sinusoïde $\sin(p\pi x)$ semblait ne pas présenter d'intérêt ; en effet, même si cela a l'avantage de restreindre l'étude à l'intervalle $]0,1[$, dans la mesure où il y a une infinité de sinusoïdes qui s'annulent dans cet intervalle, on ne fait que transformer un problème sur des données infiniment grandes en un problème sur des données infiniment petites.

Voir également http://denisevellachemla.eu/dents-de-scie.pdf pour une tentative d'explication d'une valeur moyenne de $\dfrac{1}{2}$ pour les restes modulaires vus comme des fractions rationnelles.

Interrupteurs (Denise Vella-Chemla, 31.10.2018)

A la suite d'une note publiée en juillet 2014 [1], on propose la fonction somme alternée suivante qui associe $\dfrac{1}{2}$ aux nombres premiers et des valeurs différentes de $\dfrac{1}{2}$ aux nombres composés :

$$f_D(n) = \sum_{k=2}^{n-1} \sum_{l=1}^{k} \left(cos\left(\frac{2\pi nl}{k}\right) \times (-1)^{\frac{k^2-k-2+2l}{2}} \right) - 1 + \left((-1)^{\frac{n}{2}} \times \frac{1}{2} \right)$$

Les cosinus d'angles opposés s'éliminent "presque tous" pour les nombres premiers, du fait de leur insécabilité.

Au contraire, pour les nombres composés, leur divisibilité par des nombres qui leur sont inférieurs entraîne par l'ajout des cosinus l'ajout d'un certain nombre d'unités.

Cette propriété rend la somme des cosinus :

- égale à 2 pour les nombres premiers de la forme $4k+3$,
- et égale à 1 pour les nombres premiers de la forme $4k+1$.

Le fait d'ajouter en dernier lieu au résultat $-1+(-1)^{\frac{n}{2}}$ à la somme de cosinus permet de "ramener" les images des nombres premiers égales à 2 ou 1 sur l'image $\dfrac{1}{2}$.

1. http://denise.vella.chemla.free.fr/primesommecos.pdf

Programme en $C++$ pour les sommes alternées de cosinus (les $4k+3$ ont pour image 0 et les $4k+1$ ont pour image 1)

```
#include <iostream>
#include <complex>
#include <cmath>
#include <stdio.h>

using namespace std ;
typedef complex<double> dcomp ;

int prime(int atester)
{
  unsigned long diviseur=2;
  unsigned long k = 2;
  bool pastrouve = true ;

  if (atester == 1) return 0;
  if (atester == 2) return 1;
  if (atester == 3) return 1;
  if (atester == 5) return 1;
  if (atester == 7) return 1;
  while (pastrouve)
    {
    if ((k * k) > atester) return 1;
    else
      if ((atester % k) == 0) return 0 ;
      else k++;
    }
}

int main (int argc, char* argv[])
{
  int n, i, j ;
  double somme ;
  int oppose ;
  const double PI = 4.0 * atan(1.0);

  for (n = 2 ; n <= 50 ; n++)
    {
      oppose = 1 ;
      somme = 0.0 ;
      for (i = 2 ; i <= n-1 ; i++)
  {
    std::cout << "\n" ;
    for (j = 1 ; j <= i ; j++)
      {
        oppose = (-1) * oppose ;
        somme += oppose * cos(2.0 * PI * (double) n * (double) j / (double) i) ;
        std::cout << n << "," << i << "," << j << " -> " ;
        std::cout << 360 * (double) n * (double) j / (double) i << " " ;
        std::cout << oppose * cos(2.0 * PI * (double) n * (double) j / (double) i) << "\n" ;
      }
  }
      somme = somme-1-oppose*0.5 ;
      std::cout << n << " somme globale " << somme << "\n" ;
    }
}
```

```
3,2,1 -> 540  1
3,2,2 -> 1080  1
3 somme globale 0.5

4,2,1 -> 720  -1
4,2,2 -> 1440  1

4,3,1 -> 480  0.5
4,3,2 -> 960  -0.5
4,3,3 -> 1440  -1
4 somme globale -1.5

5,2,1 -> 900  1
5,2,2 -> 1800  1

5,3,1 -> 600  0.5
5,3,2 -> 1200  -0.5
5,3,3 -> 1800  -1

5,4,1 -> 450  3.06162e-16
5,4,2 -> 900  1
5,4,3 -> 1350  -2.69484e-15
5,4,4 -> 1800  -1
5 somme globale 0.5

6,2,1 -> 1080  -1
6,2,2 -> 2160  1

6,3,1 -> 720  -1
6,3,2 -> 1440  1
6,3,3 -> 2160  -1

6,4,1 -> 540  -1
6,4,2 -> 1080  -1
6,4,3 -> 1620  -1
6,4,4 -> 2160  -1

6,5,1 -> 432  0.309017
6,5,2 -> 864  0.809017
6,5,3 -> 1296  -0.809017
6,5,4 -> 1728  -0.309017
6,5,5 -> 2160  1
6 somme globale -5.5

7,2,1 -> 1260  1
7,2,2 -> 2520  1

7,3,1 -> 840  0.5
7,3,2 -> 1680  -0.5
7,3,3 -> 2520  -1

7,4,1 -> 630  -4.28626e-16
7,4,2 -> 1260  1
7,4,3 -> 1890  -4.90478e-16
7,4,4 -> 2520  -1

7,5,1 -> 504  -0.809017
7,5,2 -> 1008  -0.309017
7,5,3 -> 1512  0.309017
7,5,4 -> 2016  0.809017
7,5,5 -> 2520  1
```

```
7,6,1 -> 420  -0.5
7,6,2 -> 840  -0.5
7,6,3 -> 1260  1
7,6,4 -> 1680  -0.5
7,6,5 -> 2100  -0.5
7,6,6 -> 2520  1
7 somme globale 0.5

8,2,1 -> 1440  -1
8,2,2 -> 2880  1

8,3,1 -> 960  0.5
8,3,2 -> 1920  -0.5
8,3,3 -> 2880  -1

8,4,1 -> 720  1
8,4,2 -> 1440  -1
8,4,3 -> 2160  1
8,4,4 -> 2880  -1

8,5,1 -> 576  -0.809017
8,5,2 -> 1152  -0.309017
8,5,3 -> 1728  0.309017
8,5,4 -> 2304  0.809017
8,5,5 -> 2880  1

8,6,1 -> 480  0.5
8,6,2 -> 960  -0.5
8,6,3 -> 1440  -1
8,6,4 -> 1920  -0.5
8,6,5 -> 2400  0.5
8,6,6 -> 2880  1

8,7,1 -> 411.429  -0.62349
8,7,2 -> 822.857  -0.222521
8,7,3 -> 1234.29  0.900969
8,7,4 -> 1645.71  -0.900969
8,7,5 -> 2057.14  0.222521
8,7,6 -> 2468.57  0.62349
8,7,7 -> 2880  -1
8 somme globale -1.5

9,2,1 -> 1620  1
9,2,2 -> 3240  1

9,3,1 -> 1080  -1
9,3,2 -> 2160  1
9,3,3 -> 3240  -1

9,4,1 -> 810  5.51091e-16
9,4,2 -> 1620  1
9,4,3 -> 2430  -3.42963e-15
9,4,4 -> 3240  -1

9,5,1 -> 648  0.309017
9,5,2 -> 1296  0.809017
9,5,3 -> 1944  -0.809017
9,5,4 -> 2592  -0.309017
9,5,5 -> 3240  1
```

```
9,6,1 -> 540  1
9,6,2 -> 1080  1
9,6,3 -> 1620  1
9,6,4 -> 2160  1
9,6,5 -> 2700  1
9,6,6 -> 3240  1

9,7,1 -> 462.857  0.222521
9,7,2 -> 925.714  -0.900969
9,7,3 -> 1388.57  -0.62349
9,7,4 -> 1851.43  0.62349
9,7,5 -> 2314.29  0.900969
9,7,6 -> 2777.14  -0.222521
9,7,7 -> 3240  -1

9,8,1 -> 405  0.707107
9,8,2 -> 810  -5.51091e-16
9,8,3 -> 1215  -0.707107
9,8,4 -> 1620  1
9,8,5 -> 2025  -0.707107
9,8,6 -> 2430  3.42963e-15
9,8,7 -> 2835  0.707107
9,8,8 -> 3240  -1
9 somme globale 6.5

10,2,1 -> 1800  -1
10,2,2 -> 3600  1

10,3,1 -> 1200  0.5
10,3,2 -> 2400  -0.5
10,3,3 -> 3600  -1

10,4,1 -> 900  -1
10,4,2 -> 1800  -1
10,4,3 -> 2700  -1
10,4,4 -> 3600  -1

10,5,1 -> 720  1
10,5,2 -> 1440  -1
10,5,3 -> 2160  1
10,5,4 -> 2880  -1
10,5,5 -> 3600  1


10,6,1 -> 600  0.5
10,6,2 -> 1200  -0.5
10,6,3 -> 1800  -1
10,6,4 -> 2400  -0.5
10,6,5 -> 3000  0.5
10,6,6 -> 3600  1

10,7,1 -> 514.286  0.900969
10,7,2 -> 1028.57  0.62349
10,7,3 -> 1542.86  0.222521
10,7,4 -> 2057.14  -0.222521
10,7,5 -> 2571.43  -0.62349
10,7,6 -> 3085.71  -0.900969
10,7,7 -> 3600  -1
```

```
10,8,1 -> 450  3.06162e-16
10,8,2 -> 900  1
10,8,3 -> 1350  -2.69484e-15
10,8,4 -> 1800  -1
10,8,5 -> 2250  -2.45548e-16
10,8,6 -> 2700  1
10,8,7 -> 3150  -3.91949e-15
10,8,8 -> 3600  -1

10,9,1 -> 400  0.766044
10,9,2 -> 800  -0.173648
10,9,3 -> 1200  -0.5
10,9,4 -> 1600  0.939693
10,9,5 -> 2000  -0.939693
10,9,6 -> 2400  0.5
10,9,7 -> 2800  0.173648
10,9,8 -> 3200  -0.766044
10,9,9 -> 3600  1
10 somme globale -5.5
```

Interrupteurs ou bien une somme alternée de cosinus assez surprenante (Denise Vella-Chemla, 4.11.2018)

A la suite d'une note publiée en juillet 2014 [1], on propose la fonction somme alternée suivante qui associe $\frac{1}{2}$ aux nombres premiers et des valeurs différentes de $\frac{1}{2}$ aux nombres composés :

$$f_D(n) = \sum_{k=2}^{n-1} \sum_{l=1}^{k} \left(cos\left(\frac{2\pi nl}{k}\right) \times (-1)^{\frac{k^2-k-2+2l}{2}} \right) - 1 + \left((-1)^{\frac{n}{2}} \times \frac{1}{2} \right)$$

Les cosinus d'angles opposés s'éliminent "presque tous" pour les nombres premiers, du fait de leur insécabilité.

Au contraire, pour les nombres composés, leur divisibilité par des nombres qui leur sont inférieurs entraîne par l'ajout des cosinus l'ajout d'un certain nombre d'unités.

Cette propriété rend la somme des cosinus :

- égale à 2 pour les nombres premiers de la forme $4k+3$,
- et égale à 1 pour les nombres premiers de la forme $4k+1$.

Le fait d'ajouter en dernier lieu au résultat $-1+\left((-1)^{\frac{n}{2}} \times \frac{1}{2}\right)$ à la somme de cosinus permet de "ramener" les images des nombres premiers égales à 2 ou 1 sur l'image $\frac{1}{2}$.

On fait calculer par programme cette somme alternée de cosinus pour les entiers jusqu'à 10000 [2].

Cette somme alternée présente la propriété suivante : pour les puissances de nombres premiers, la fonction $f_D(n)$ coïncide avec des fonctions affines ; ainsi, $f_D(9) = 6.5, f_D(27) = 24.5, f_D(81) = 78.5$, i.e. $f_D(x = 3^k) = x - 2.5$; ou bien $f_D(25) = 10.5, f_D(125) = 60.5, f_D(625) = 310.5, f_D(3125) = 1560.5$, i.e. $f_D(x = 5^k) = \frac{1}{2}x - 2$; etc.

On trouve la formule générale pour les puissances de premiers $x = p^k$: si on note $f_D(x) = ax - b$, a est égal à $\frac{2}{p-1}$ et b est égal à $\frac{3p+1}{2p-2}$.

Pour les nombres dont la factorisation fait intervenir plusieurs nombres premiers différents, on n'arrive pas à dégager de formule générale qui coïnciderait avec la somme alternée de cosinus qu'on a proposée.

On est intrigué par ces résultats et on revient à l'idée initiale qui consistait à calculer la somme suivante :

$$g_D(n) = \sum_{k=2}^{n-1} \sum_{l=1}^{k} cos\left(\frac{2\pi nl}{k}\right)$$

On fait calculer à nouveau par programme cette somme de cosinus pour les entiers jusqu'à 1000 [3].

Les images des puissances de nombres premiers obéissent à la formule :

$$g_D(p^k) = \frac{p^k - p}{p-1}$$

Les images des produits simples de nombres premiers (à la puissance 1) obéissent à la formule :

$$g_D(pq) = p + q$$

1. http://denise.vella.chemla.free.fr/primesommecos.pdf
2. Le programme de calcul de $f_D(n)$ peut être trouvé ici http://denise.vella.chemla.free.fr/trouveformuleoppose.pdf. Le résultat du programme peut être trouvé ici http://denise.vella.chemla.free.fr/affin.pdf
3. Le programme de calcul de $g_D(n)$ peut être trouvé ici http://denise.vella.chemla.free.fr/sumsumcos.pdf. Le résultat du programme peut être trouvé ici http://denise.vella.chemla.free.fr/ressumsumcos.pdf

Par exemple, $g_D(21) = g_D(3 \times 7) = 3 + 7 = 10$ ou bien $g_D(10) = g_D(2 \times 5) = 2 + 5 = 7$.

Le calcul de $g_D(210) = g_D(2\times3\times5\times7) = 2+3+5+7+6+10+14+15+21+35+30+42+70+105 = 365$.

La fonction $g_D(n)$ est ainsi très simple à calculer pour les produits de nombres premiers simples.
Pour les produits faisant intervenir des puissances, on peut appliquer la formule :

$$g_D(p^k \times x) = (p + 1)g_D(p^{k-1} \times x).$$

Conjecture de Goldbach, où l'on retrouve ζ autrement (Denise Vella-Chemla, 26.1.2019)

On s'intéresse à la conjecture de Goldbach qui stipule que tout nombre pair supérieur strictement à 2 est la somme de deux nombres premiers.

On rappelle qu'un nombre premier inférieur à $\dfrac{n}{2}$, qui ne partage aucun de ses restes avec n un nombre pair supérieur à 2, dans toute division par un nombre premier inférieur à $\sqrt{n}$, est un décomposant de Goldbach de n.

En effet, si x inférieur à $\dfrac{n}{2}$ ne partage aucun de ses restes avec n dans toute division par un nombre premier inférieur à $\sqrt{n}$, alors $n-x$ est lui aussi premier.

La probabilité qu'un nombre x inférieur à $\dfrac{n}{2}$ soit premier est fournie par le théorème des nombres premiers ; elle vaut :

$$\frac{\frac{n}{2}}{ln\left(\frac{n}{2}\right)}$$

Supposons maintenant que x est premier. Etudions le non-partage d'un reste au moins entre x et n dans les divisions par les nombres premiers inférieurs à $\sqrt{n}$.

Puisque x est premier, on sait au moins qu'il n'a aucun reste nul dans toute division par un nombre premier inférieur à $\sqrt{n}$.

Dans une division par 3, il lui reste 2 possibilités de reste (1 et 2), et il a une chance sur deux (i.e. 1/2) d'obtenir l'un ou l'autre.

Dans une division par 5, il lui reste 4 possibilités de reste (1, 2, 3 ou 4), et il a une chance sur 4 (i.e. 1/4) d'obtenir l'un ou l'autre.

Dans une division par 7, il lui reste 6 possibilités de reste (1, 2, 3, 4, 5 et 6), et il a une chance sur 6 (i.e. 1/6) d'obtenir l'un ou l'autre.

Plus généralement, dans une division par p, il lui reste $p-1$ possibilités de reste $(1, 2, \ldots,\ p-1)$, et il a une chance sur $p-1$ (i.e. 1/(p-1)) d'obtenir l'un ou l'autre.

Tous ces événements ayant des probabilités indépendantes, la probabilité d'obtenir leur conjonction est le produit des probabilités de chaque événement séparé (les événements considérés étant "x et n ont même reste dans une division par 3", "x et n ont même reste dans une division par 5", etc.).

Ce produit s'écrit :

$$\prod_{p\ premier\ <\sqrt{n}} \frac{1}{p-1}$$

On peut le réécrire :

$$\prod_{p\ premier\ <\sqrt{n}} \frac{1}{p^{-(-1)}-1}$$

puis

$$-\prod_{p\ premier\ <\sqrt{n}} \frac{1}{1-p^{-(-1)}}$$

et l'on reconnaît alors $-\zeta(-1)$. Ramanujan a démontré que $\zeta(-1) = -\dfrac{1}{12}$. La note [1] fournit une démonstration simple de ce fait.

On obtient donc comme probabilité globale qu'un nombre x soit d'une part premier, et d'autre part ne partage aucun de ses restes avec n dans une division par un nombre premier inférieur à $\sqrt{n}$ [2] :

$$\frac{\frac{n}{2}}{ln\left(\frac{n}{2}\right)} \times (-\zeta(-1))$$

soit :

$$\frac{n}{2\ ln\ n - 2\ ln\ 2} \times \frac{1}{12}.$$

Ceci semble rendre la conjecture de Goldach vraie à partir de $n = 92$ [3].

1. Par définition $S = 1 + 2 + 3 + 4 + 5 + ...$ On remarque qu'en faisant la différence terme à terme :

$$\begin{array}{rccccl} S - B = & 1+2 & +3+4 & +5+6 & ... & \\ & -1+2 & -3+4 & -5+6 & ... & \\ = & 0+4 & +0+8 & +0+12 & ... & = 4(1+2+3+...) = 4S \end{array}$$

Donc $S - 4S = B$, i.e. $-3S = B$, d'où $S = -\dfrac{B}{3} = -\dfrac{\frac{1}{4}}{3}$ Ainsi, on retrouve le résultat attendu : $S = -\dfrac{1}{12}$.

2. Le fait pour x de ne partager aucun reste avec n dans les divisions par les nombres premiers inférieurs à $\sqrt{n}$ n'a rien à voir avec le fait d'être premier à n. Cette condition est nécessaire (i.e. *impliquée*) mais non suffisante (i.e. *impliquante*). Par exemple, 17 et 81, dont la somme vaut 98, sont tous les deux premiers à 98, mais ils n'en sont pas pour autant des décomposants de Goldbach (de 98) puisque 17 partage le reste de 2 avec 98 lorsqu'on les divise par 3 (Gauss écrit cela $17 \equiv 98\ (mod\ 3)$, c'est lui qui a attiré l'attention de tous sur l'importance de travailler dans les corps premiers).

3. $\dfrac{92}{2\ ln\ 92 - 2\ ln\ 2}.\dfrac{1}{12} = 1.0012254835$ alors que $\dfrac{90}{2\ ln\ 90 - 2\ ln\ 2)}.\dfrac{1}{12} = 0.9851149163$.

Conjecture de Goldbach, où l'on retrouve ζ autrement (Denise Vella-Chemla, 29.5.2019)

On s'intéresse à la conjecture de Goldbach qui stipule que tout nombre pair supérieur strictement à 2 est la somme de deux nombres premiers.

On rappelle qu'un nombre premier inférieur à $\dfrac{n}{2}$, qui ne partage aucun de ses restes avec n un nombre pair supérieur à 2, dans toute division par un nombre premier inférieur à $\sqrt{n}$, est un décomposant de Goldbach de n.

En effet, si x inférieur à $\dfrac{n}{2}$ ne partage aucun de ses restes avec n dans toute division par un nombre premier inférieur à $\sqrt{n}$, alors $n-x$ est lui aussi premier.

La probabilité asymptotique qu'un nombre x inférieur à $\dfrac{n}{2}$ soit premier est fournie par le théorème des nombres premiers ; elle vaut :

$$\frac{\dfrac{n}{2}}{ln\left(\dfrac{n}{2}\right)}$$

La minoration de $\pi(k)$ (le nombre de nombres premiers inférieurs à k) par $\dfrac{k}{ln\ k}$ est fournie dans [1], page 69, pour $x \geq 17$.

Supposons maintenant que x est premier. Etudions les probabilités d'égalité des restes de x et n quand on les divise par les nombres premiers inférieurs à $\sqrt{n}$.

Puisqu'on a supposé x premier, on sait au moins qu'il n'a aucun reste nul lorsqu'on le divise par un nombre premier inférieur à $\sqrt{n}$.

n a un certain reste, lorsqu'on le divise par un nombre premier inférieur à $\sqrt{n}$. x doit "éviter" le reste en question (ne doit pas avoir le même).

Si on considère une division de n par l'un de ses diviseurs premiers p de reste nul, x n'a que ce reste nul (0) à éviter. Or x a déjà évité 0 par le fait qu'il est premier. x a le choix entre $p-1$ restes possibles dans la division par p.

Si on considère une division de n par un nombre premier qui n'est pas l'un de ses diviseurs, n a selon ce nombre premier un reste non-nul que x doit éviter. x a alors le choix entre $p-2$ restes possibles dans sa division par p, qu'il peut avoir à égales probabilités l'un ou l'autre mais on va utiliser le fait que $\dfrac{1}{p-2} > \dfrac{1}{p-1}$ et minorer chaque probabilité selon un nombre premier p donné par $\dfrac{1}{p-1}$, pour homogénéiser les différents cas (considération des diviseurs ou des non-diviseurs de n).

Voyons des exemples, pour fixer les idées : dans une division par 3, on minore le nombre de possibilités par 2 possibilités de reste (1 et 2), et x a une chance sur deux (i.e. 1/2) d'obtenir l'un ou l'autre.

Dans une division par 5, il reste à x 4 possibilités de reste (1, 2, 3 ou 4), et x a une chance sur 4 (i.e. 1/4) d'obtenir l'un ou l'autre.

Dans une division par 7, il reste à x 6 possibilités de reste (1, 2, 3, 4, 5 et 6), et x a une chance sur 6 (i.e. 1/6) d'obtenir l'un ou l'autre.

Plus généralement, dans une division par p, on minore la probabilité que x et n aient le même reste ainsi : il y a $p-1$ possibilités de restes possibles au maximum pour x (qui sont $1, 2, \ldots, \ p-1$), et x a une chance sur $p-1$ (i.e. 1/(p-1)) d'obtenir l'un ou l'autre de ces restes.

Tous ces événements ayant des probabilités indépendantes, la probabilité d'obtenir leur conjonction est le produit des probabilités de chaque événement séparé (les événements considérés étant "x et n ont même reste dans une division par 3", "x et n ont même reste dans une division par 5", etc.).

Ce produit s'écrit :

$$\prod_{p\ premier\ <\sqrt{n}} \frac{1}{p-1}$$

On peut le réécrire :

$$\prod_{p\ premier\ <\sqrt{n}} \frac{1}{p^{-(-1)}-1}$$

puis

$$-\prod_{p\ premier\ <\sqrt{n}} \frac{1}{1-p^{-(-1)}}$$

On peut étendre ce produit à l'infinité des nombres premiers car en fait, c'est selon tout nombre premier que n et x ne peuvent être congrus, pour que le complémentaire à n de x qu'est $n-x$ soit premier. On reconnaît alors $-\zeta(-1)$ dans le produit proposé pour que x et n aient des restes différents dans un division par un nombre premier quelconque. Ramanujan a démontré que $\zeta(-1) = -\dfrac{1}{12}$. La note [1]fournit une démonstration simple de ce fait.

On obtient donc comme probabilité globale qu'un nombre x soit d'une part premier, et d'autre part ne partage aucun de ses restes avec n dans une division par un nombre premier inférieur à $\sqrt{n}$ (en fait quelconque) [2] :

$$\frac{\frac{n}{2}}{ln\left(\frac{n}{2}\right)} \times (-\zeta(-1))$$

soit :

$$\frac{n}{2\ ln\ n - 2\ ln\ 2} \times \frac{1}{12}.$$

Ceci semble rendre la conjecture de Goldach vraie à partir de $n = 92$ [3].

Tentative de réécriture mathématique :

On cherche à démontrer que $\forall n$ pair, $\exists x, 3 \leq x \leq n/2$ premier impair tel que $n-x$ est premier aussi.

$$\begin{array}{lllll} (1) & x \text{ premier} & \Longleftrightarrow & \forall p \text{ premier} \leq \sqrt{x}, & x \not\equiv 0\ (mod\ p). \\ (2) & n-x \text{ premier} & \Longleftrightarrow & \forall p \text{ premier} \leq \sqrt{n-x}, & n-x \not\equiv 0\ (mod\ p) \\ & & \Longleftrightarrow & \forall p \text{ premier} \leq \sqrt{n-x}, & x \not\equiv n\ (mod\ p). \end{array}$$

On peut remplacer dans (1) la condition $\forall p$ premier $\leq \sqrt{x}$ par la condition plus forte $\forall p$ premier $\leq \sqrt{n/2}$ puisqu'on a posé $x \leq n/2$.

1. Par définition $S = 1+2+3+4+5+....$ On remarque qu'en faisant la différence terme à terme :

$$\begin{array}{rcllll} S - B = & & 1+2 & +3+4 & +5+6 & ... \\ & & -1+2 & -3+4 & -5+6 & ... \\ & = & 0+4 & +0+8 & +0+12 & ... = 4(1+2+3+...) = 4S \end{array}$$

Donc $S - 4S = B$, i.e. $-3S = B$, d'où $S = -\dfrac{B}{3} = -\dfrac{\frac{1}{4}}{3}$ Ainsi, on retrouve le résultat attendu : $S = -\dfrac{1}{12}$.

2. Le fait pour x de ne partager aucun reste avec n dans les divisions par les nombres premiers inférieurs à $\sqrt{n}$ n'a rien à voir avec le fait d'être premier à n. Cette condition est nécessaire (i.e. *impliquée*) mais non suffisante (i.e. *impliquante*). Par exemple, 17 et 81, dont la somme vaut 98, sont tous les deux premiers à 98, mais ils n'en sont pas pour autant des décomposants de Goldbach (de 98) puisque 17 partage le reste de 2 avec 98 lorsqu'on les divise par 3 (Gauss écrit cela $17 \equiv 98\ (mod\ 3)$, c'est lui qui a attiré l'attention de tous sur l'importance de travailler dans les corps premiers).

3. $\dfrac{92}{2\ ln\ 92 - 2\ ln\ 2}.\dfrac{1}{12} = 1.0012254835$ alors que $\dfrac{90}{2\ ln\ 90 - 2\ ln\ 2)}.\dfrac{1}{12} = 0.9851149163.$

On minore le nombre de nombres premiers inférieurs à $\frac{n}{2}$ par $\dfrac{\frac{n}{2}}{log\left(\frac{n}{2}\right)}$.

Il s'agit alors de trouver combien de nombres dans cet ensemble de nombres premiers inférieurs à $\frac{n}{2}$, dont on a le cardinal, partagent leur reste avec n ; le partage d'un reste avec n le nombre pair considéré consiste à "fixer" le reste possible et donc à diminuer le nombre de restes possibles de 1 selon chaque module ; on doit multiplier le cardinal $\pi\left(\frac{n}{2}\right)$ minoré par $\dfrac{\frac{n}{2}}{log\left(\frac{n}{2}\right)}$ (qui correspond à la condition (1) ci-dessus) par la probabilité qu'il y ait un partage de reste selon chaque nombre premier indépendamment (qui correspond à la condition (2) ci-dessus) et cette probabilité a comme valeur $-\zeta(-1) = \frac{1}{12}$. C'est un cardinal d'ensemble qu'on obtient par ce procédé de multiplication d'un cardinal par une probabilité. Un tel calcul semble faire sens et assure un cardinal d'au moins 1 à partir de 92.

Bibliographie

[1] J. B. Rosser et L. Schoenfeld, *Approximate formulas for some functions of prime numbers*, dedicated to Hans Rademacher for his seventieth birthday, Illinois J. Math., Volume 6, Issue 1 (1962), 64-94.

Annexe à la proposition denitac.pdf (Denise Vella-Chemla, 23.7.2019)

On cherche à décomposer un nombre pair n en somme de 2 nombres premiers $p_1 + p_2$.

On ne peut pas faire référence à $\zeta(-1)$ comme on l'a fait dans [1]. On peut cependant, pour obtenir une minoration du nombre de décomposants de Goldbach de n, utiliser le cardinal $\left|\mathcal{P}_{\frac{n}{2}}\right|$ de l'ensemble des nombres premiers inférieurs ou égaux à $\frac{n}{2}$ et le multiplier par le produit $\prod\limits_{p \leq \sqrt{n}} \left(1 - \frac{1}{p}\right)$ qui compte combien de chances a le nombre premier p_1 de ne pas partager son reste avec n selon chaque module p inférieur à $\sqrt{n}$ (le fait de ne pas partager son reste avec n permet à p_1 d'avoir un complémentaire à n (appelé p_2) qui est premier également).

La minoration [1] de $\pi(x)$ (le nombre de nombres premiers inférieurs à x) par $\frac{x}{\log x}$ est fournie dans [2], page 69, pour $x \geq 17$ (Corollaire 1, (3.5), du Théorème 2, dont la démonstration est fournie au paragraphe 7 de [2]).

On a en conséquence $|\mathcal{P}_{\frac{n}{2}}| > \frac{\frac{n}{2}}{\log\left(\frac{n}{2}\right)}$.

La minoration de $\prod\limits_{p \leq \sqrt{n}} \left(1 - \frac{1}{p}\right)$ est également fournie dans [2], page 70 (c'est le corollaire (3.27) du Théorème 7 dont la démonstration est fournie au paragraphe 8 de [2], avec γ la constante d'Euler-Mascheroni).

$$\text{(3.27)} \qquad \frac{e^{-\gamma}}{\log x}\left(1 - \frac{1}{\log^2 x}\right) < \prod_{p \leq x}\left(1 - \frac{1}{p}\right) \quad \text{pour } 1 < x.$$

En multipliant ces expressions ensemble, on obtient que le nombre de décomposants de Goldbach de n doit être supérieur à :

$$\frac{n/2}{\log(n/2)} \frac{e^{-\gamma}}{\log\sqrt{n}}\left(1 - \frac{1}{\log^2\sqrt{n}}\right)$$

qui est strictement supérieur à 1 à partir de 24.

Bibliographie

[1] http://denisevellachemla.eu/denitac.pdf.

[2] J. B. Rosser et L. Schoenfeld, *Approximate formulas for some functions of prime numbers*, dedicated to Hans Rademacher for his seventieth birthday, Illinois J. Math., Volume 6, Issue 1 (1962), 64-94.

1. Cette minoration est à distinguer du Théorème des nombres premiers, prouvé indépendamment par Hadamard et La Vallée-Poussin, et qui fournit une tendance asymptotique pour $\pi(x)$.

Réécrire

Denise Vella-Chemla (7.12.2019) aidée par Leila Schneps pour la section 1

1. Caractérisation des décomposants de Goldbach d'un nombre pair

Soit n un nombre pair supérieur à 4 et p_k un nombre premier compris entre 3 et $\sqrt{n}$.

Notons $F(p_k, n) = \{m \in \mathbb{N} : m \text{ impair}, \sqrt{n} \leq m \leq n/2, m \neq 0\ [p_k], m \neq n\ [p_k]\}$

Appelons $D(n) = \cap F(p_k, n)$ l'intersection des ensembles $F(p_k, n)$ pour tous les premiers p_k compris entre 3 et $\sqrt{n}$.

Démontrons que $D(n)$ ne contient que des nombres premiers.

Lemme 1 : Soit m un entier positif impair. Si m n'est divisible par aucun nombre premier compris entre 3 et $\sqrt{m}$, alors m est premier.

Démonstration : Supposons que m ne soit pas premier. Alors il existe un nombre premier $p < m$ qui divise m. Mais on sait que p n'est pas compris entre 3 et $\sqrt{m}$, donc $p > \sqrt{m}$. On pose $k = m/p$. On a donc $kp = m$. Si $k \geq \sqrt{m}$, alors puisqu'on a aussi $p > \sqrt{m}$, on obtient $kp > m$, ce qui est impossible. On doit donc avoir $k < \sqrt{m}$. Mais comme tout entier, l'entier k est divisible par un nombre premier $q \leq k$. Comme q divise k et k divise m, on a que q divise aussi m, et comme $k \leq \sqrt{m}$, on a que $q \leq \sqrt{m}$, ce qui contredit notre hypothèse de départ que m n'est divisible par aucun premier $\leq \sqrt{m}$. QED.

Appliquons ce résultat à $D(n)$ pour obtenir que $D(n)$ ne contient que des nombres premiers.

Lemme 2 : $D(n)$ ne contient que des nombres premiers *.

Démonstration : Soit $m \in D(n)$. Alors m est impair et $m \leq n/2$. On sait par le lemme 1 que si m n'est divisible par aucun premier compris entre 3 et $\sqrt{m}$, alors m est premier. Mais par la définition de $D(n)$, on sait déjà que m n'est divisible par aucun premier compris entre 3 et $\sqrt{n}$, et puisque $m < n$, on a $\sqrt{m} < \sqrt{n}$ et donc a fortiori m n'est divisible par aucun premier compris entre 3 et $\sqrt{m}$, donc par le lemme 1, m est bien premier. QED.

Lemme 3 : Si m appartient à $D(n)$, alors $n - m$ est premier.

Démonstration : On commence par montrer qu'aucun nombre premier p compris entre 3 et $\sqrt{n}$ ne divise $n - m$. En effet, si $n - m$ est divisible par p, alors m est congru à n modulo p, ce qui contredit le fait que m appartient à $D(n)$. Ensuite, on note que puisque $n - m < n$, on a $\sqrt{n-m} < \sqrt{n}$ et donc a fortiori, $n-m$ n'est divisible par aucun premier $\leq \sqrt{n-m}$, donc par le lemme 1, $n-m$ est bien un nombre premier.

Si $D(n)$ est non vide, alors n vérifie la conjecture de Goldbach.

2. Existence d'un décomposant de Goldbach pour tout nombre pair

On a vu que $D(n)$ ne contient que des nombres premiers qui sont décomposants de Goldbach de n. Il faut maintenant démontrer que $D(n)$ est non vide pour que n vérifie la conjecture de Goldbach.

Essayons de comprendre pourquoi $D(n) = \cap F(p_k, n)$ ne peut être vide. On reprend l'écriture initiale qu'on avait choisie, sous forme logique : dire que l'intersection des ensembles de la forme $\{\neg 0_{p_k} \wedge \neg n_{p_k}\}$

$*$. si $D(n)$ est vide, le lemme est vrai par vacuité.

Preuve de Victor Varin que les nombres premiers annulent ma somme de somme de cosinus et qu'ils sont les seuls nombres à le faire (Denise Vella-Chemla, 13.7.2017)

$$dn1 = \sigma(n) = \sum_{k=1}^{n}\sum_{l=1}^{k} \cos\frac{2\pi nl}{k}.$$

$\sigma(n)$ est la somme des diviseurs de n.

$\sigma(1) = 1$
$\sigma(2) = 3$
$\sigma(3) = 4$

Maintenant la belle formule ($ssc = sumsumcos$).

$$dn2 = ssc(n) = \sum_{k=2}^{n-1}\sum_{l=1}^{k} \cos\frac{2\pi nl}{k}.$$

$ssc(2) = 0$
$ssc(3) = 0$
$ssc(4) = 2$
$ssc(5) = 0$

On a :

$$\sum_{k=1}^{n}\sum_{l=1}^{k} \cos\frac{2\pi nl}{k} = \sum_{l=1}^{1} \cos 2\pi nl + \sum_{k=2}^{n-1}\sum_{l=1}^{k} \cos\frac{2\pi nl}{k} + \sum_{l=1}^{n} \cos 2\pi l$$

$$\begin{aligned} dn3 \quad &= \sigma(n) = 1 + n + \sum_{k=2}^{n-1}\sum_{l=1}^{k} \cos\frac{2\pi nl}{k} \\ &= 1 + n + ssc(n) \end{aligned}$$

Ce qui est évident pour les nombres premiers.

Maintenant prouvons formellement que les nombres premiers annulent la fonction $ssc(n)$.

On définit $sss(n)$ par :

$$dn4 = \sum_{k=1}^{n}\sum_{l=1}^{k} \sin\frac{2\pi nl}{k}$$

$sss(2) = 0$
$sss(3) = 0$
$sss(4) = 0$
$sss(5) = 0$

Mais nous n'utiliserons pas cette propriété, elle découlera automatiquement.

Maintenant prenons

$v(n) = ssc(n) + i.sss(n)$ avec $i^2 = -1$

$$dn5 = v(n) = \sum_{k=2}^{n-1}\sum_{l=1}^{k} \cos\frac{2\pi nl}{k} + i\left(\sum_{k=2}^{n-1}\sum_{l=1}^{k} \sin\frac{2\pi nl}{k}\right)$$

De façon triviale, $Re(v(n)) = ssc(n)$ quel que soit $sss(n)$! Mais :

$$dn6 = v(n) = \sum_{k=2}^{n-1}\sum_{l=1}^{k} e^{\frac{2i\pi nl}{k}}$$

$v(4) = 2$

Maintenant, considérons la somme intérieure $dn6$:

$$dn7 = sm(n) = \sum_{l=1}^{k} e^{\frac{2i\pi nl}{k}}$$

Maintenant, une ruse importante : oublions temporairement que n est un entier dans $dn7$!!

$$dn7 = sm(n) = \sum_{l=1}^{k} \left(e^{\frac{2i\pi n}{k}}\right)^{l}$$

Et posons :

$$e^{\frac{2i\pi n}{k}} = X$$

alors

$$dn7 = sm(n) = \sum_{l=1}^{k} X^l = \frac{X(X^k - 1)}{X - 1}$$

Finalement

$$\begin{aligned} dn7 &= sm(n) &= \frac{e^{\frac{2i\pi n}{k}}\left(\left(e^{\frac{2i\pi n}{k}}\right)^k - 1\right)}{e^{\frac{2i\pi n}{k}} - 1} \\ &= \frac{e^{\frac{2i\pi n}{k}}\left(e^{2i\pi n} - 1\right)}{e^{\frac{2i\pi n}{k}} - 1} \end{aligned}$$

Ces calculs sont valides si le dénominateur est non nul !

Maintenant, si n est premier, alors $\frac{n}{k}$ est fractionnaire et le dénominateur n'est pas nul, dans la mesure où $exp(2i\pi n) = 1$ pour n entier !

Ainsi la preuve est faite que les nombres premiers n annulent $ssc(n)$. Mais si n est un nombre composé, il faut utiliser une petite ruse : considérons seulement $k|n$ (k divise n) dans $dn6$.

subs(n=d*k, dn7) mais d est légèrement différente d'une valeur entière et alors la formule ci-dessous a du sens :

$$sm(dk) = \frac{e^{2i\pi d}(e^{2i\pi dk} - 1)}{e^{2i\pi d} - 1}$$
$$e^{2i\pi d} = Y$$
$$e^{2i\pi d} = 1$$
$$\frac{Y^k - 1}{Y - 1} = \sum_{j=0}^{k-1} Y^j$$
$$sm(dk) = e^{2i\pi d} \sum_{j=0}^{k-1} \left(e^{2i\pi d}\right)^j$$

On rappelle alors que d est entier et que donc,

$$sm(dk) = k$$

ce qui termine la preuve.

Orthogonalité

- On note $\mathcal{F}(x) = \{y \in \mathbb{N}^\times \mid 3 \leq y \leq x\}$
- On note $98^\perp$ l'ensemble des décomposants de Goldbach de 98.
- $98 \in (2\mathbb{N}) \cap (3\mathbb{N}+2) \cap (5\mathbb{N}+3) \cap (7\mathbb{N})$.
- $98^\perp \in \ \mathcal{F}(49) \ \cap$ $[(2\mathbb{N}+1) \cap (3\mathbb{N}+1) \cap [(5\mathbb{N}+1) \cup (5\mathbb{N}+2) \cup (5\mathbb{N}+4)] \cap [(7\mathbb{N}+1) \cup \ldots \cup (7\mathbb{N}+6)]]$.

  Attention : Bien avoir à l'esprit que les règles de développement de la multiplication $\cap$ sur l'addition $\cup$ se font comme habituellement en algèbre, i.e. l'union $(7\mathbb{N}+1) \cup \ldots \cup (7\mathbb{N}+6)$ ne représente pas tous les entiers sauf les multiples de 7 par exemple. On développe par distributivité d'une opération sur l'autre.

- Plus généralement, si on note $\mathcal{P}$ l'ensemble des nombres premiers.
- $n \in \displaystyle\bigcap_{a \in \mathcal{F}(\sqrt{n}) \ \cap \ \mathcal{P}^\times, b \in \{0,\ldots,a-1\}} \{ax+b\}$ et
- $n^\perp \in \mathcal{F}(n/2) \ \cap \displaystyle\bigcap_{a \in \mathcal{F}(\sqrt{n}) \ \cap \ \mathcal{P}^\times, b' \in \{1,\ldots,a-1\}} \{ax+b', \text{ avec } b' \neq b, \forall a\}$.

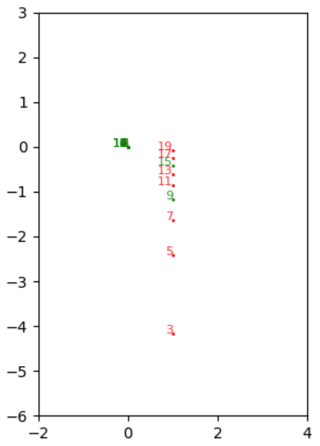


Fig. 1 : $x \leq 20$

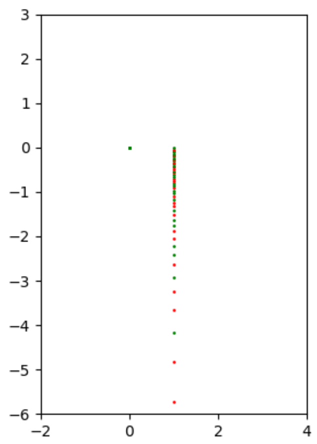
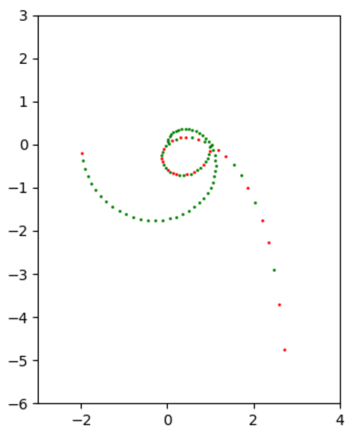


Fig. 2 : $x \leq 100$



$Fig.\,3$: 100 nbs, exposants $\leq$ 98

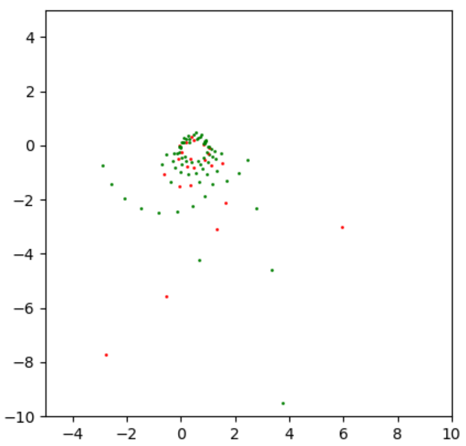


Fig. 4

L'explication de l'alignement de tous les nombres (si ce n'est un point qu'on semble distinguer comme non aligné avec les autres) sur la droite de partie réelle 1 est la suivante :

Puisque $t_k = e^{\frac{-i\pi k}{n}}$ et $z_k = \dfrac{1-t_k^m}{1-t_k}$, avec $k \in [1, n-1]$, si $m = n$, alors on a $t_k^m = e^{-i\pi k}$ et de ce fait,

$$\begin{aligned} z_k &= \frac{1-e^{-i\pi k}}{1-e^{\frac{-i\pi k}{n}}} \\ &= \frac{1-(-1)^k}{1-e^{\frac{-i\pi k}{n}}} \end{aligned}$$

Le dernier numérateur ci-dessus est nul lorsque k est pair.

Si k est impair, on a :

$$\begin{aligned} z_k &= \frac{2}{1-e^{\frac{-i\pi k}{n}}} \\ &= \frac{2\,\left(1-e^{\frac{i\pi k}{n}}\right)}{\left(1-\cos\,\frac{k\pi}{n}\right)^2+\left(\sin\,\frac{k\pi}{n}\right)^2} \end{aligned}$$

$$
\begin{aligned}
z_k &= \frac{2\left(1-e^{\frac{i\pi k}{n}}\right)}{2-2\cos\frac{k\pi}{n}} \\
&= \frac{1-e^{\frac{i\pi k}{n}}}{1-\cos\frac{k\pi}{n}} \\
&= \frac{1-\left(\cos\frac{k\pi}{n}+i\sin\frac{k\pi}{n}\right)}{1-\cos\frac{k\pi}{n}} \\
&= \frac{1-\cos\frac{k\pi}{n}}{1-\cos\frac{k\pi}{n}} + i\frac{\sin\frac{k\pi}{n}}{1-\cos\frac{k\pi}{n}}
\end{aligned}
$$

qui est de partie réelle constante égale à 1, d'où l'alignement des complexes obtenus comme images des nombres entiers.

La dernière figure ci-dessous montre les nombres pairs mal lisibles et tous collés sur 0. Pour ne pas écraser davantage les nombres les uns sur les autres, on a laissé de côté le nombre 1 qui est bien aligné aussi mais tout en bas à $-32i$.

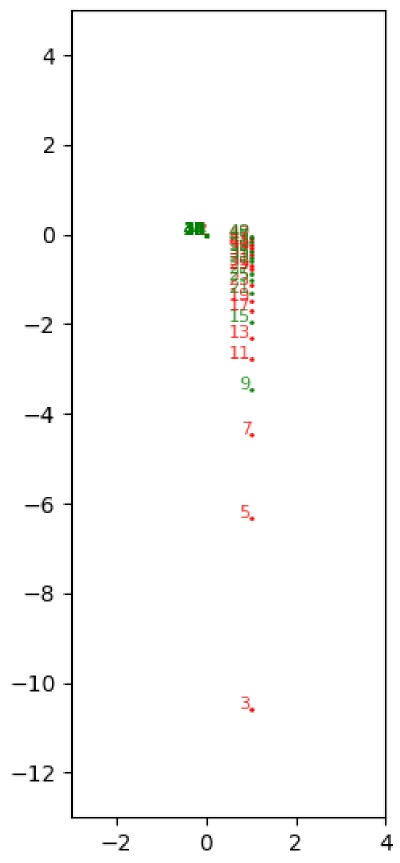


Fig. 4

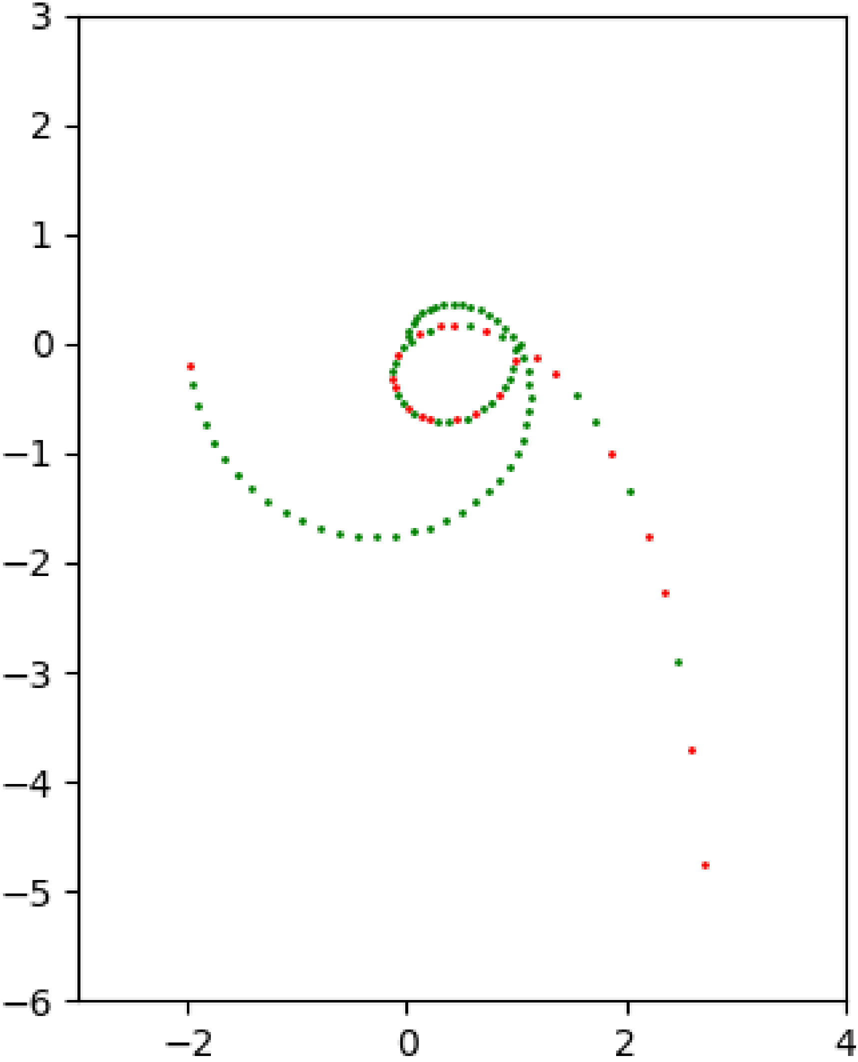
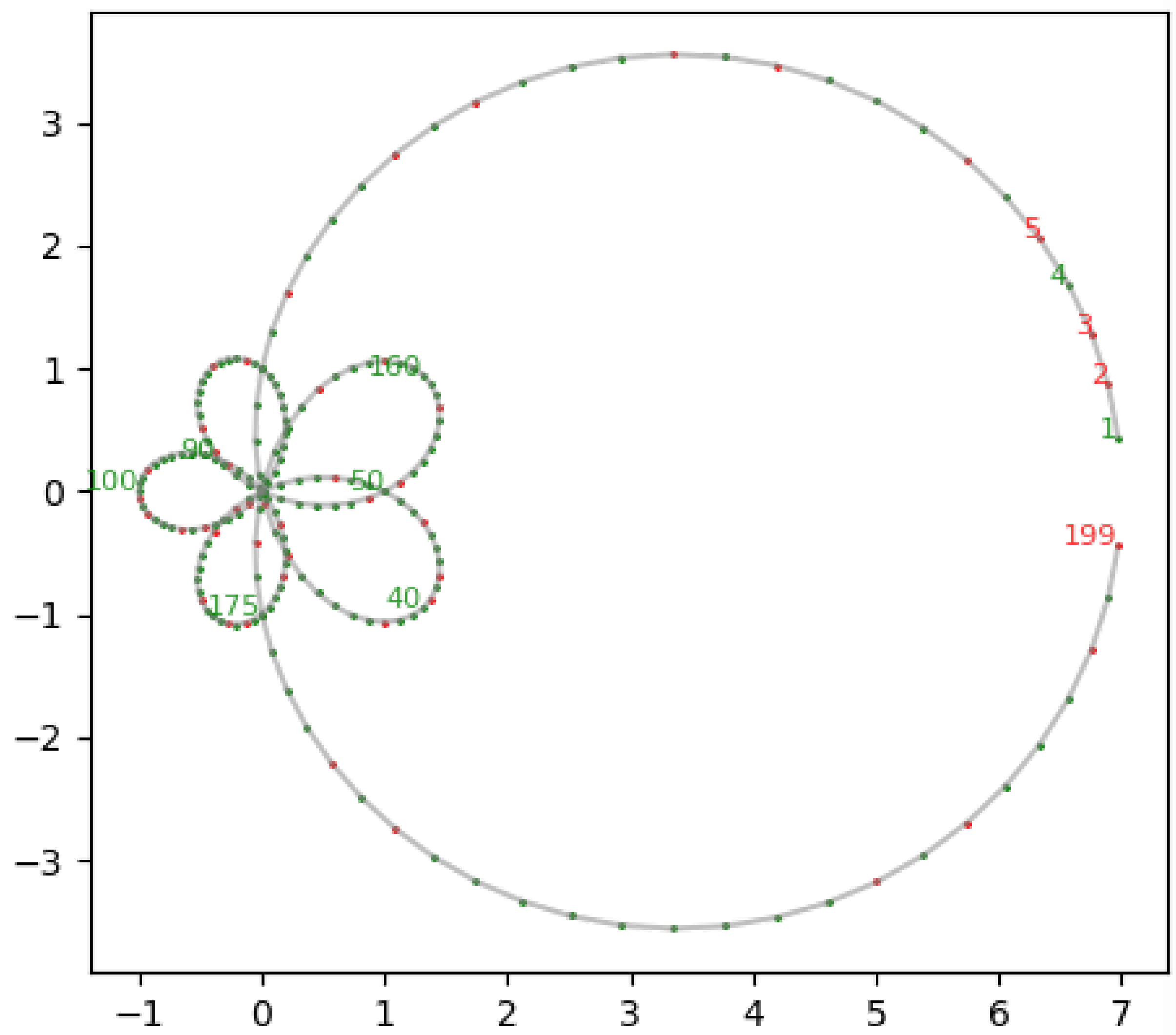

3
2
1
0
−1
−2
−3
−4
−5
−6
−2
0
2
4

Figure 1



emacs25@vellachemla-X510UA

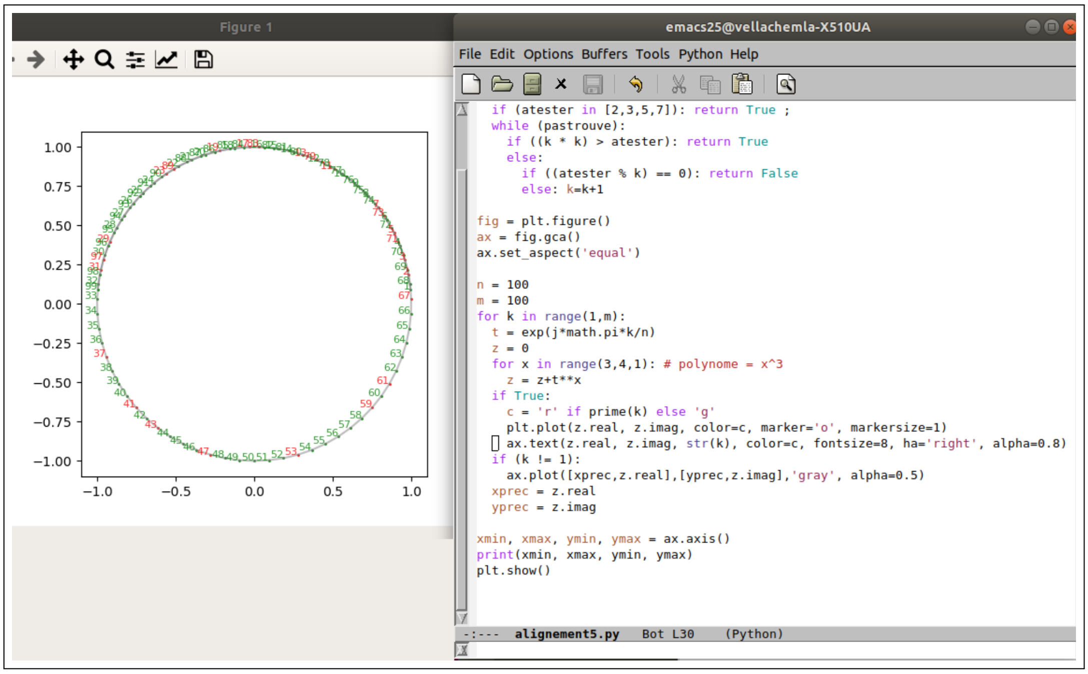
File Edit Options Buffers Tools Python Help

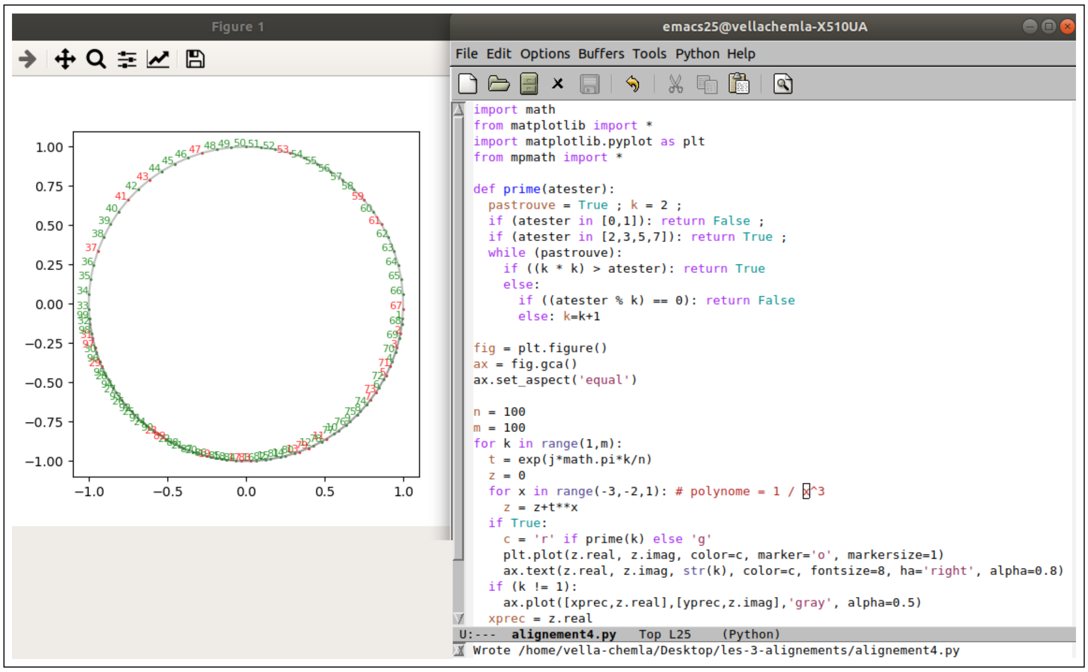
```
import math
from matplotlib import *
import matplotlib.pyplot as plt
from mpmath import *

def prime(atester):
  pastrouve = True ; k = 2 ;
  if (atester in [0,1]): return False ;
  if (atester in [2,3,5,7]): return True ;
  while (pastrouve):
    if ((k * k) > atester): return True
    else:
      if ((atester % k) == 0): return False
      else: k=k+1

fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')

n = 100
m = 200
for k in range(1,m):
  t = exp(j*math.pi*k/n)
  z = 1/t+1+t+t*t+t*t*t+t*t*t*t+t*t*t*t*t
  if True:
    c = 'r' if prime(k) else 'g'
    plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
    if (k <= 5) or (k == 90) or (k == 160) or (k == 40) or (k == 100) or (k == 175) or (k == 50) or (k == 199):
      ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
  if (k != 1):
    ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
```

-:--- fleur5petales.py Top L28 (Python)

Wrote /home/vella-chemla/Desktop/fleur5petales.py

Inversions (Denise Vella-Chemla, 8.10.2020)





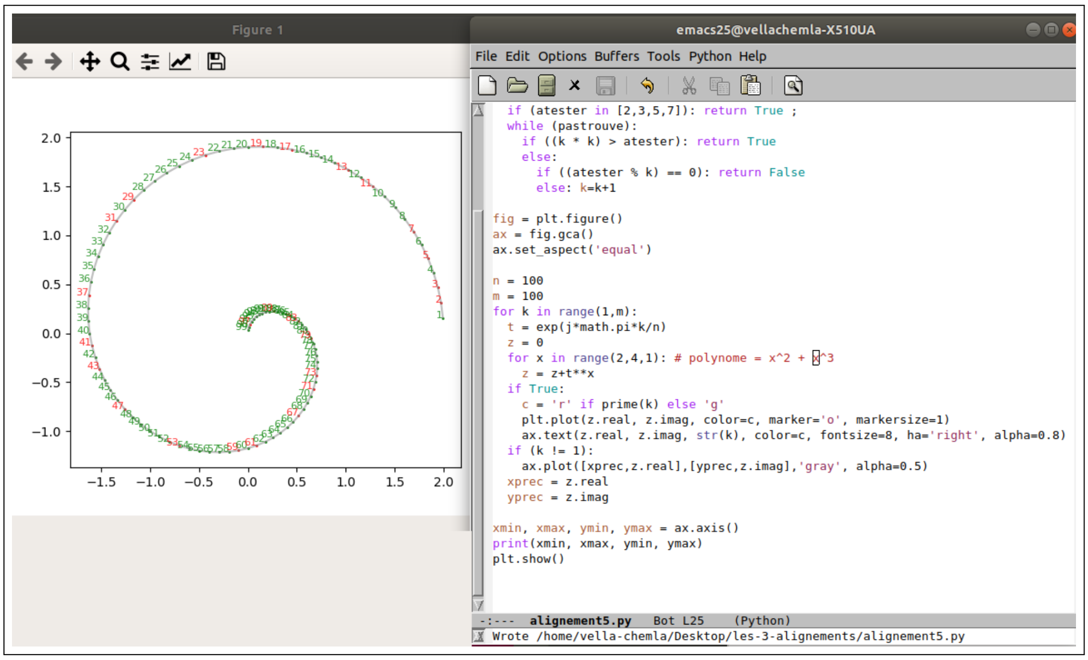


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
  if (atester in [2,3,5,7]): return True ;
  while (pastrouve):
    if ((k * k) > atester): return True
    else:
      if ((atester % k) == 0): return False
      else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
  t = exp(j*math.pi*k/n)
  z = 0
  for x in range(2,4,1): # polynome = x^2 + x^3
    z = z+t**x
  if True:
    c = 'r' if prime(k) else 'g'
    plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
    ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
  if (k != 1):
    ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
  xprec = z.real
  yprec = z.imag
xmin, xmax, ymin, ymax = ax.axis()
print(xmin, xmax, ymin, ymax)
plt.show()
-:--- alignement5.py Bot L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement5.py

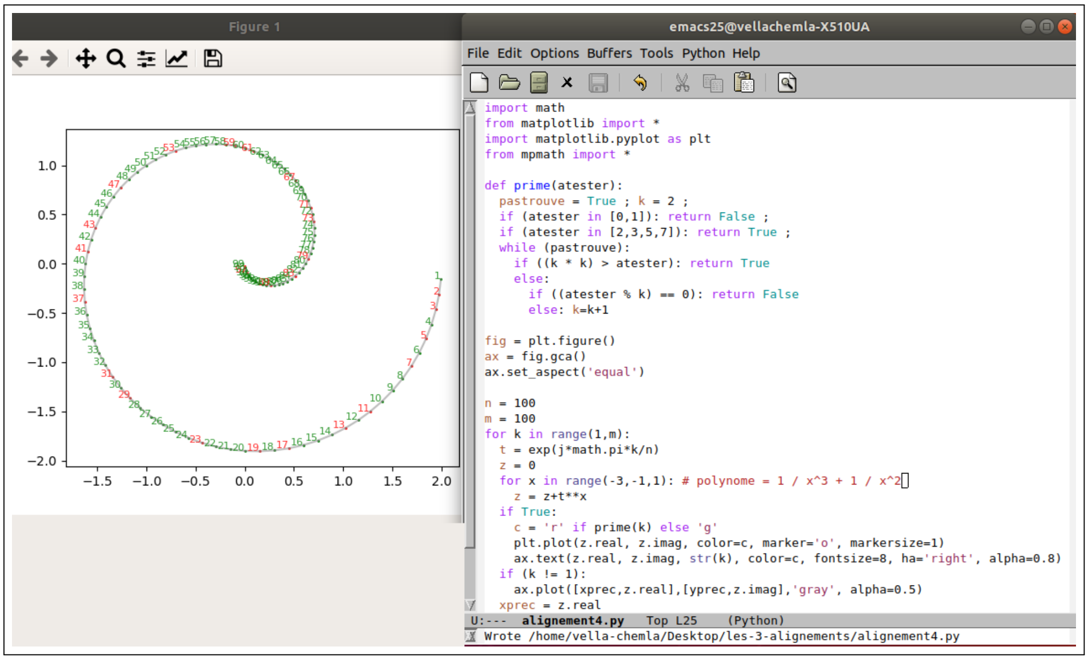


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
import math
from matplotlib import *
import matplotlib.pyplot as plt
from mpmath import *
def prime(atester):
  pastrouve = True ; k = 2 ;
  if (atester in [0,1]): return False ;
  if (atester in [2,3,5,7]): return True ;
  while (pastrouve):
    if ((k * k) > atester): return True
    else:
      if ((atester % k) == 0): return False
      else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
  t = exp(j*math.pi*k/n)
  z = 0
  for x in range(-3,-1,1): # polynome = 1 / x^3 + 1 / x^2
    z = z+t**x
  if True:
    c = 'r' if prime(k) else 'g'
    plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
    ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
  if (k != 1):
    ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
  xprec = z.real
U:--- alignement4.py Top L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement4.py

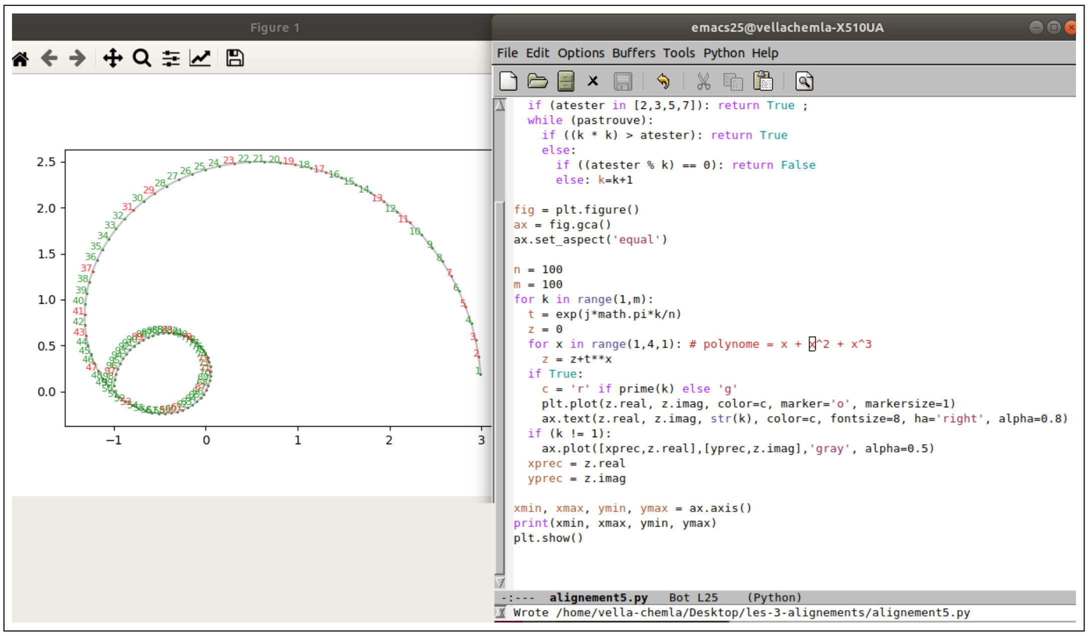


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
  if (atester in [2,3,5,7]): return True ;
  while (pastrouve):
    if ((k * k) > atester): return True
    else:
      if ((atester % k) == 0): return False
      else: k=k+1

fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')

n = 100
m = 100
for k in range(1,m):
  t = exp(j*math.pi*k/n)
  z = 0
  for x in range(1,4,1): # polynome = x + x^2 + x^3
    z = z+t**x
  if True:
    c = 'r' if prime(k) else 'g'
    plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
    ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
  if (k != 1):
    ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
  xprec = z.real
  yprec = z.imag

xmin, xmax, ymin, ymax = ax.axis()
print(xmin, xmax, ymin, ymax)
plt.show()
-:--- alignement5.py   Bot L25   (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement5.py

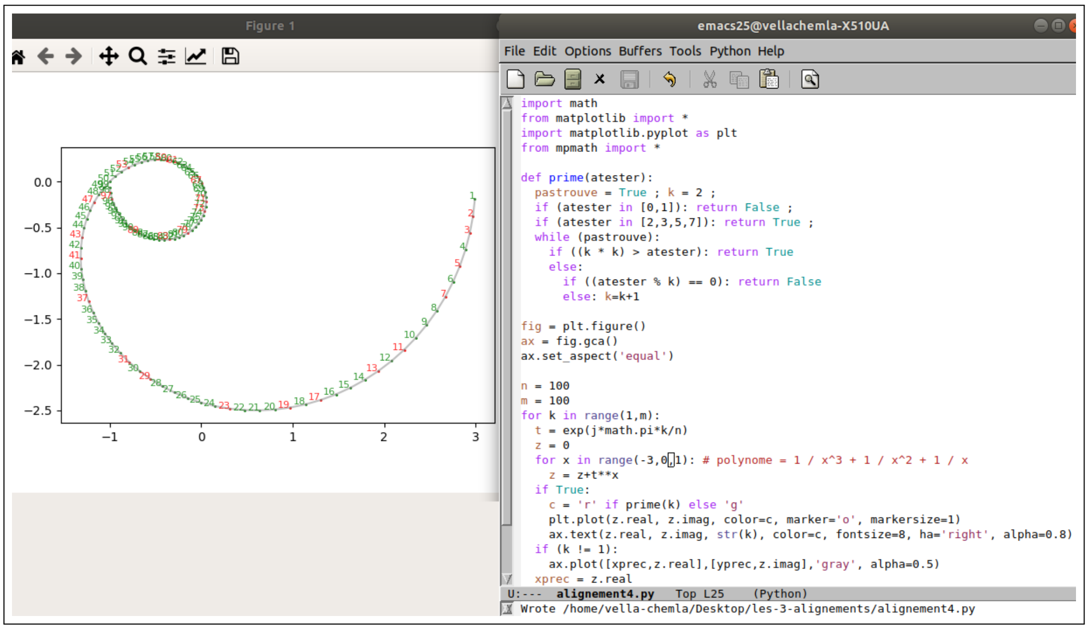


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
import math
from matplotlib import *
import matplotlib.pyplot as plt
from mpmath import *

def prime(atester):
  pastrouve = True ; k = 2 ;
  if (atester in [0,1]): return False ;
  if (atester in [2,3,5,7]): return True ;
  while (pastrouve):
    if ((k * k) > atester): return True
    else:
      if ((atester % k) == 0): return False
      else: k=k+1

fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')

n = 100
m = 100
for k in range(1,m):
  t = exp(j*math.pi*k/n)
  z = 0
  for x in range(-3,0,1): # polynome = 1 / x^3 + 1 / x^2 + 1 / x
    z = z+t**x
  if True:
    c = 'r' if prime(k) else 'g'
    plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
    ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
  if (k != 1):
    ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
  xprec = z.real
U:--- alignement4.py   Top L25   (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement4.py

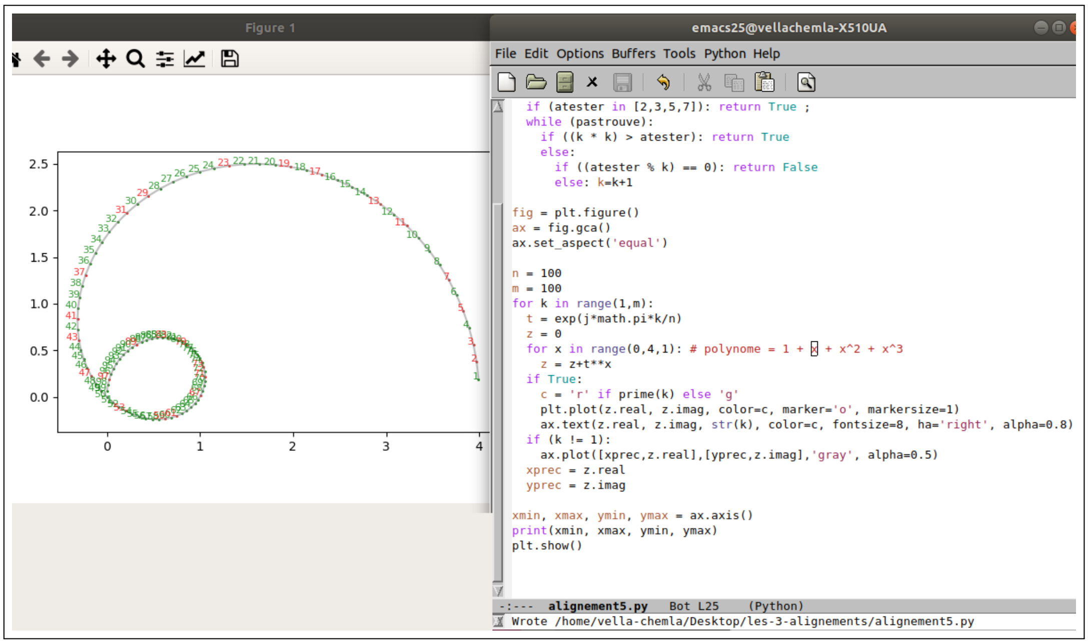


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
2.5
2.0
1.5
1.0
0.5
0.0
0
1
2
3
4
  if (atester in [2,3,5,7]): return True ;
  while (pastrouve):
    if ((k * k) > atester): return True
    else:
      if ((atester % k) == 0): return False
      else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
  t = exp(j*math.pi*k/n)
  z = 0
  for x in range(0,4,1): # polynome = 1 + x + x^2 + x^3
    z = z+t**x
  if True:
    c = 'r' if prime(k) else 'g'
    plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
    ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
  if (k != 1):
    ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
  xprec = z.real
  yprec = z.imag
xmin, xmax, ymin, ymax = ax.axis()
print(xmin, xmax, ymin, ymax)
plt.show()
-:--- alignement5.py Bot L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement5.py

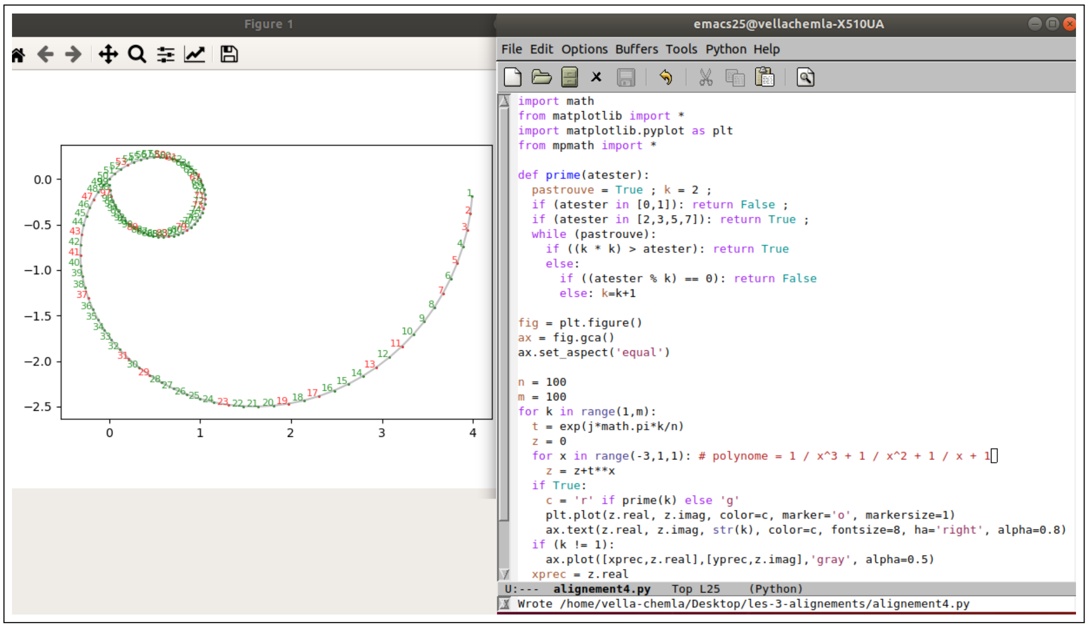


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
import math
from matplotlib import *
import matplotlib.pyplot as plt
from mpmath import *
def prime(atester):
  pastrouve = True ; k = 2 ;
  if (atester in [0,1]): return False ;
  if (atester in [2,3,5,7]): return True ;
  while (pastrouve):
    if ((k * k) > atester): return True
    else:
      if ((atester % k) == 0): return False
      else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
  t = exp(j*math.pi*k/n)
  z = 0
  for x in range(-3,1,1): # polynome = 1 / x^3 + 1 / x^2 + 1 / x + 1
    z = z+t**x
  if True:
    c = 'r' if prime(k) else 'g'
    plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
    ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
  if (k != 1):
    ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
  xprec = z.real
U:--- alignement4.py Top L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement4.py

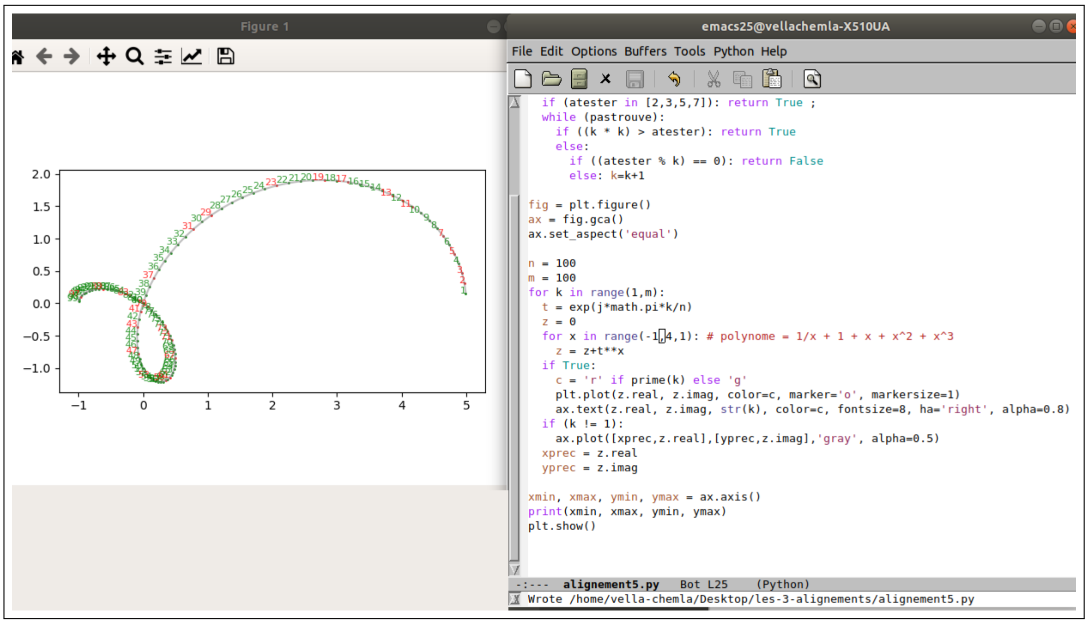


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
if (atester in [2,3,5,7]): return True ;
while (pastrouve):
if ((k * k) > atester): return True
else:
if ((atester % k) == 0): return False
else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
t = exp(j*math.pi*k/n)
z = 0
for x in range(-1,4,1): # polynome = 1/x + 1 + x + x^2 + x^3
z = z+t**x
if True:
c = 'r' if prime(k) else 'g'
plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
if (k != 1):
ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
xprec = z.real
yprec = z.imag
xmin, xmax, ymin, ymax = ax.axis()
print(xmin, xmax, ymin, ymax)
plt.show()
-:--- alignement5.py Bot L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement5.py

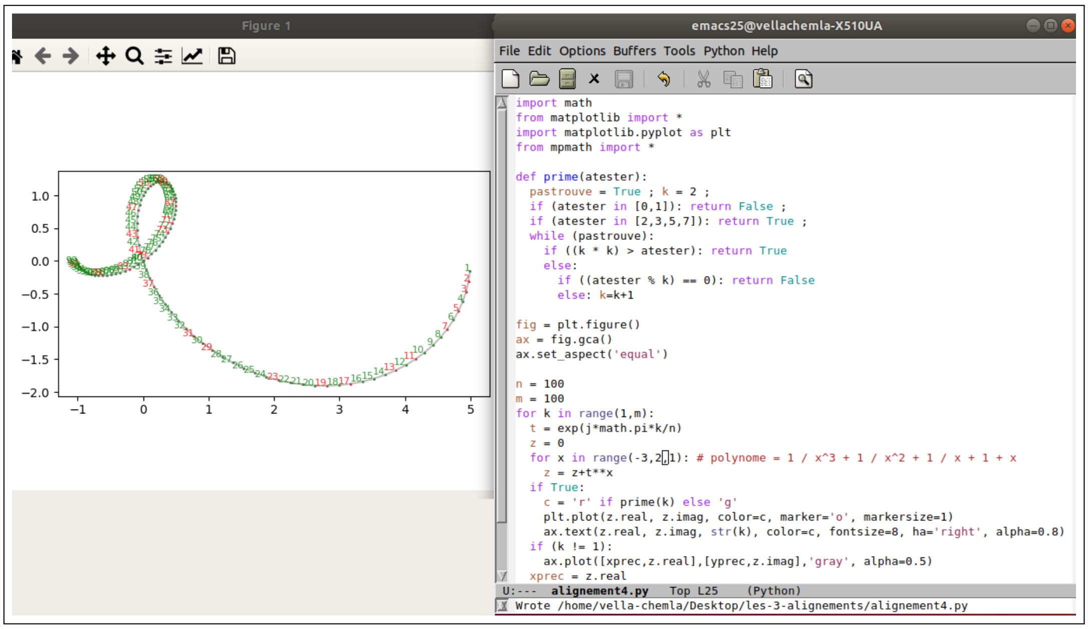


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
import math
from matplotlib import *
import matplotlib.pyplot as plt
from mpmath import *
def prime(atester):
pastrouve = True ; k = 2 ;
if (atester in [0,1]): return False ;
if (atester in [2,3,5,7]): return True ;
while (pastrouve):
if ((k * k) > atester): return True
else:
if ((atester % k) == 0): return False
else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
t = exp(j*math.pi*k/n)
z = 0
for x in range(-3,2,1): # polynome = 1 / x^3 + 1 / x^2 + 1 / x + 1 + x
z = z+t**x
if True:
c = 'r' if prime(k) else 'g'
plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
if (k != 1):
ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
xprec = z.real
U:--- alignement4.py Top L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement4.py

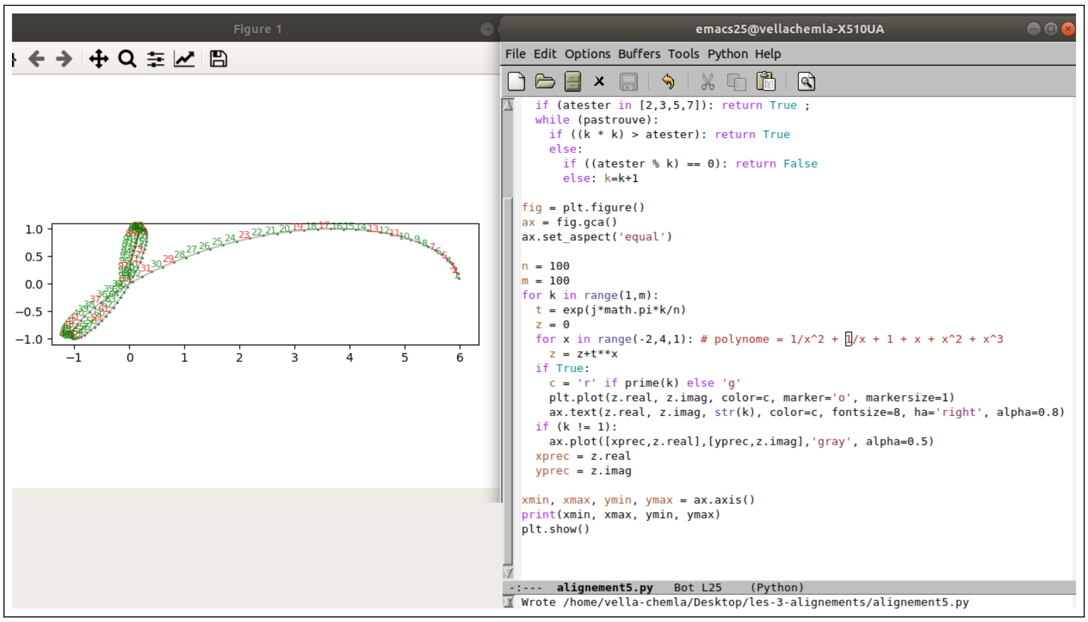


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
if (atester in [2,3,5,7]): return True ;
while (pastrouve):
if ((k * k) > atester): return True
else:
if ((atester % k) == 0): return False
else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
t = exp(j*math.pi*k/n)
z = 0
for x in range(-2,4,1): # polynome = 1/x^2 + 1/x + 1 + x + x^2 + x^3
z = z+t**x
if True:
c = 'r' if prime(k) else 'g'
plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
if (k != 1):
ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
xprec = z.real
yprec = z.imag
xmin, xmax, ymin, ymax = ax.axis()
print(xmin, xmax, ymin, ymax)
plt.show()
-:--- alignement5.py Bot L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement5.py

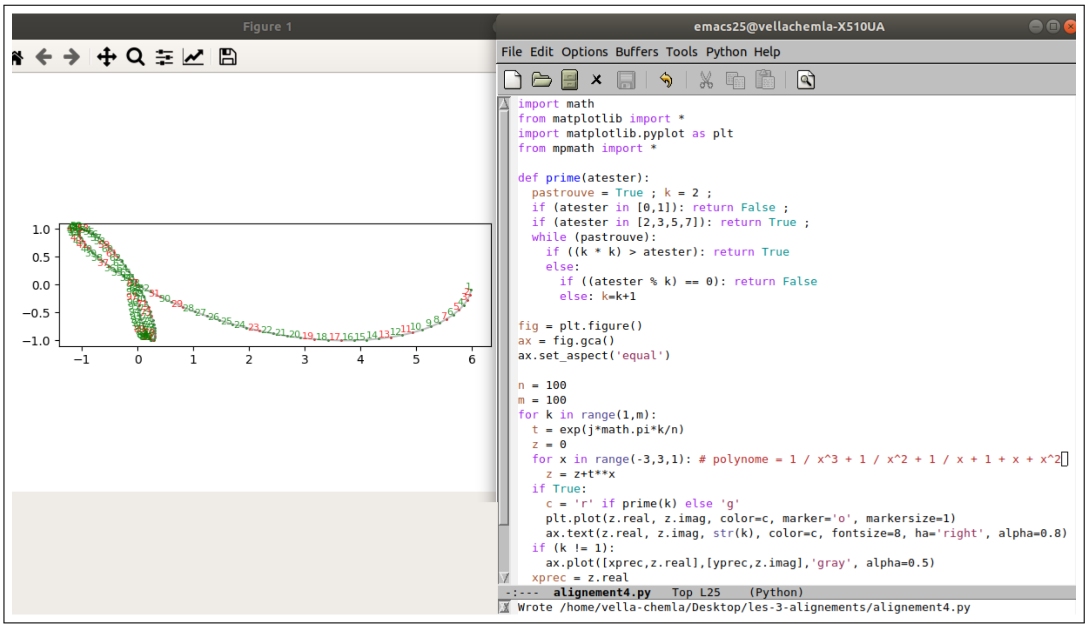


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
import math
from matplotlib import *
import matplotlib.pyplot as plt
from mpmath import *
def prime(atester):
pastrouve = True ; k = 2 ;
if (atester in [0,1]): return False ;
if (atester in [2,3,5,7]): return True ;
while (pastrouve):
if ((k * k) > atester): return True
else:
if ((atester % k) == 0): return False
else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
t = exp(j*math.pi*k/n)
z = 0
for x in range(-3,3,1): # polynome = 1 / x^3 + 1 / x^2 + 1 / x + 1 + x + x^2
z = z+t**x
if True:
c = 'r' if prime(k) else 'g'
plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
if (k != 1):
ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
xprec = z.real
-:--- alignement4.py Top L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement4.py

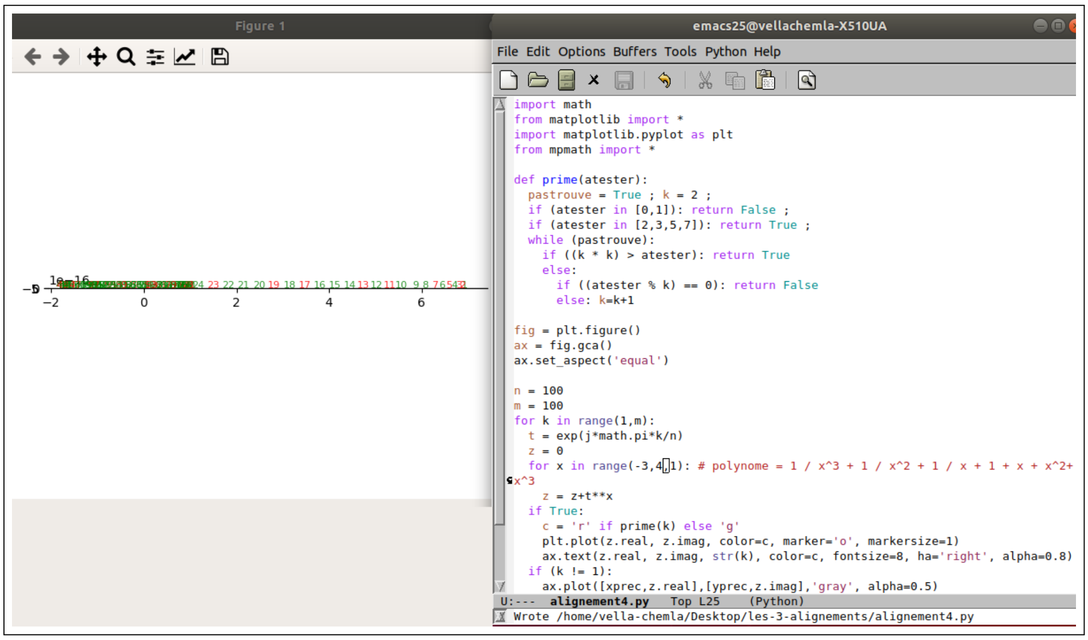


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
import math
from matplotlib import *
import matplotlib.pyplot as plt
from mpmath import *
def prime(atester):
  pastrouve = True ; k = 2 ;
  if (atester in [0,1]): return False ;
  if (atester in [2,3,5,7]): return True ;
  while (pastrouve):
    if ((k * k) > atester): return True
    else:
      if ((atester % k) == 0): return False
      else: k=k+1
fig = plt.figure()
ax = fig.gca()
ax.set_aspect('equal')
n = 100
m = 100
for k in range(1,m):
  t = exp(j*math.pi*k/n)
  z = 0
  for x in range(-3,4,1): # polynome = 1 / x^3 + 1 / x^2 + 1 / x + 1 + x + x^2+x^3
    z = z+t**x
  if True:
    c = 'r' if prime(k) else 'g'
    plt.plot(z.real, z.imag, color=c, marker='o', markersize=1)
    ax.text(z.real, z.imag, str(k), color=c, fontsize=8, ha='right', alpha=0.8)
  if (k != 1):
    ax.plot([xprec,z.real],[yprec,z.imag],'gray', alpha=0.5)
U:--- alignement4.py Top L25 (Python)
Wrote /home/vella-chemla/Desktop/les-3-alignements/alignement4.py

Affectée (Denise Vella-Chemla, Toussaint 2020)

Je voudrais ici revenir sur des programmes qui m'ont plu, même s'ils sont inefficaces. Je les apprécie plutôt parce qu'ils me rappellent une idée, un concept, que j'ai réussi à implémenter. Ci-après, deux façons de mettre en œuvre le crible d'Eratosthène pour compter ou écrire les nombres premiers ; le premier programme est frugal, il n'utilise essentiellement qu'une ligne :

```python
import time
import math
from math import sqrt

tps1 = time.time()
def premiers(n):
    return [p for p in range(2,n) if all([p%q for q in range(2,int(sqrt(p)))])]
pix=premiers(1000)
print(pix)
print("pix "+str(len(pix)))
print("Temps d execution : %s secondes..." % (time.time()-tps1))
```

Le second programme est basé sur la formule de Legendre, qui fait intervenir la fonction de Möbius ; il s'agit d'appliquer le principe d'inclusion/exclusion (pour exprimer grossièrement l'idée, on enlève les multiples de 2, les multiples de 3, mais ayant de ce fait enlevé deux fois au lieu d'une seule fois les multiples de 6, qui sont à la fois des multiples de 2 et des multiples de 3, on rajoute le nombre des multiples de 6, etc...). L'idée derrière le second programme est la suivante : on prend l'ensemble des nombres premiers jusqu'à la racine de n, et on fabrique à partir des éléments de cet ensemble tous les produits possibles dont tous les facteurs sont différents. S'il y a k éléments dans l'ensemble, on peut utiliser, pour identifier chaque produit à calculer, un mot de k booléens qui dit pour chaque élément s'il est l'un des facteurs du produit en cours de calcul ou pas. Un raffinement supplémentaire consiste à utiliser les codes de Gray (algorithme de Knuth) pour énumérer tous les mots booléens par le "flip" d'un seul bit pour passer d'un mot à un autre. La formule de calcul du nombre de nombres premiers inférieurs ou égaux à x, notée $\pi(x)$ s'écrit alors $\pi(x) = \pi(\sqrt{x}) + \sum\limits_{d|P} \mu(d) \left\lfloor \frac{x}{d} \right\rfloor - 1$ avec $P = \prod\limits_{p \leq \sqrt{x}} p$.

La fonction de Möbius, $\mu(d)$, multiplicative, vaut -1 pour le produit d'un nombre impair de nombres premiers tous différents, elle vaut 1 pour le produit d'un nombre pair de nombres premiers tous différents, et elle vaut 0 sinon, sauf pour 1 pour lequel elle vaut 1 ($\mu(1) = 1$).

```python
import bisect, math, time
primes = [2]
def next_prime(n):
    while any([n % d == 0 for d in primes]): n += 1
    return n
def add_primes(n):
    r = math.sqrt(n)
    while primes[-1] < r: primes.append(next_prime(primes[-1] + 1))
def pi(n):
    # l'algorithme G de Knuth (code binaire de Gray) est utilis\'{e} pour g\'{e}n\'{e}rer
    # tous les sous-ensembles d'un ensemble
    m = bisect.bisect_right(primes, math.sqrt(n))
    a = [0 for _ in range(m)]
    s, mu = m, 1
    while (True):
        d = math.prod([primes[i] for i in range(m) if a[i]])
        s, mu = s + mu * math.floor(n/d), -mu
        j = 0 if mu == -1 else a.index(1) + 1
        if j == m: break
        a[j] = 1 - a[j]
    return s

add_primes(10000)
for k in range(1, 11):
    n = 1000*k
    t0 = time.time(); p = pi(n); t1 = time.time()
    print('n = {:5d}, pi = {:4d}, time = {:10.6f} s'.format(n, p, t1-t0))
```

On peut aussi trouver les nombres premiers en utilisant une formule récurrente pour le calcul de la somme des diviseurs fournie par Euler dans son article *Découverte d'une loi tout extraordinaire des nombres par rapport à la somme de leurs diviseurs* [1]. Plutôt que d'implémenter l'algorithme tel que proposé par Euler [2], on peut utiliser la formule récurrente de la séquence A000203 de calcul de la somme des diviseurs, fournie par D. Giard dans la partie FORMULA en bas de page, sur le site de l'OEIS [3] qui se programme par exemple ainsi :

```
#include <iostream>

int main (int argc, char* argv[]) {
  int n, k, somme ;
  int sigma[120] ;

  sigma[1] = 1 ;
  for (n = 2 ; n <= 100 ; ++n) {
    somme = 0 ;
    for (k=1 ; k < n ; k++)
      somme = somme+(-(n*n)+5*k*n-5*k*k)*sigma[k]*sigma[n-k] ;
    sigma[n] = (12*somme)/(n*n*(n-1)) ;
    if (sigma[n] == n+1)
      std::cout << n << "  " ;
  }
}
```

On peut aussi utiliser le fait que Gauss a démontré qu'un nombre premier p, contrairement à un nombre composé, est caractérisé par le fait qu'exactement la moitié (i.e. un ensemble de cardinal $\frac{p-1}{2}$) des nombres strictement compris entre 0 et p sont des résidus quadratiques de p (sont des carrés modulo p). On calcule les carrés successifs en ajoutant les nombres impairs successifs [4], et on marque les nombres qui sont effectivement des carrés modulaires dans un tableau de booléens.

```
import time
import numpy

pix = 0
tps1 = time.time()
nmax = 1000
marque = numpy.zeros(nmax)
for p in range(2,nmax):
      nbsol = 0
      somme = 0
      for x in range(1,p):
          marque[x] = False
      for x in range(0,int((p-1)/2)):
          somme = (somme +(2*x+1)) % p ;
          marque[somme] = True
      for x in range(1,p):
        if (marque[x]):
          nbsol = nbsol+1
      if (nbsol == (p-1)/2):
          print (p,end=' ')
          pix = pix+1
print("pix "+str(pix))
print("Temps d execution : %s secondes..." % (time.time()-tps1))
```

On peut aussi utiliser le fait que les divisions entières de n par tous les nombres qui lui sont inférieurs "tombent pile juste sur le quotient d'après" lorsqu'il y a divisibilité et dans ces cas seulement. Ce qui fait qu'un nombre premier, exactement selon la manière dont on le définit, n'a que 2 divisions qui "tombent juste", notamment relativement au nombre qui le précède, ce qui fait que la différence entre leurs deux sommes de quotients entiers est égale à 2.

1. L. EULER. *Découverte d'une loi tout extraordinaire des nombres par rapport à la somme de leurs diviseurs.* Éd. Commentationes arithmeticae 2, p.639, 1849.
2. On trouve le programme initial, écrit en décembre 2006, ici http://denisevellachemla.eu/Algo-d-Euler-somme-div-Decouv-loi.pdf ou en annexe de http://denise.vella.chemla.free.fr/noel2006.pdf.
3. Online Encyclopedia of Integer Sequences, suite A000203, https://oeis.org/A000203.
4. https://fr.wikipedia.org/wiki/Preuve$_{s}ans_{m}ots$.

```python
import time

tps1=time.time()
pix = 0
somme = 0
for x in range(1,1001):
    sommeprec = somme
    somme = 0
    for k in range(1,x+1):
        somme = somme+int(x/k)
    if (somme-sommeprec == 2):
        print(x, end=' ')
        pix=pix+1
print("pix "+str(pix))
print("Temps d execution : %s secondes..." % (time.time()-tps1))
```

Enfin, voici un programme que j'affectionne particulièrement, même s'il est totalement inefficace : il s'agit d'utiliser des matrices circulantes. Il faut voir le fait de "barrer un nombre sur k lors de l'exécution du crible d'Eratosthène" comme équivalent au fait de calculer les puissances d'une matrice circulante de taille $k \times k$, dont des sous-matrices reviennent périodiquement, identiques à elles-mêmes, lorsqu'on élève la matrice initiale aux différentes puissances de 2 à k.

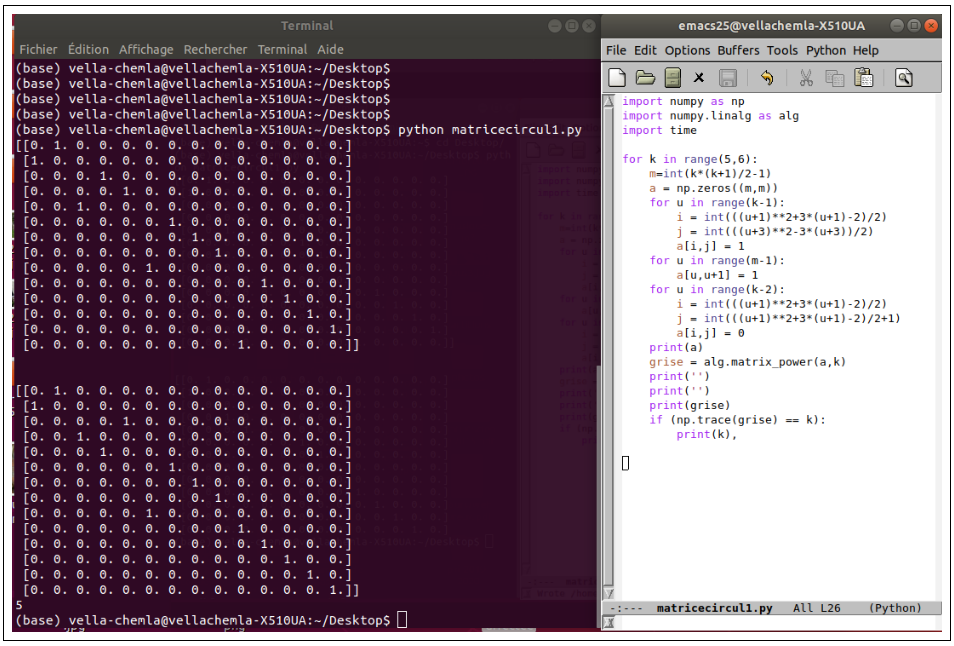
Voici la forme générale de la matrice G.

$$\begin{pmatrix}
0 & 1 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
1 & 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & 0 & 1 & 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & 0 & 0 & 1 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & 1 & 0 & 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & \dots & 0 & 1 & 0 & 0 & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & \dots & 0 & 0 & 1 & 0 & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & \dots & 0 & 0 & 0 & 1 & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & \dots & 1 & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 & 1 & 0 & 0 & 0 & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 & 0 & 1 & 0 & 0 & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 & 0 & 0 & 1 & 0 & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 & 0 & 0 & 0 & 1 & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 1 & 0 & 0 & 0 & 0 & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots
\end{pmatrix}$$

Tous les $\dots$ sont des 0.

Selon cette modélisation, un nombre k s'avère premier si, lorsqu'on élève "sa" matrice de la forme ci-dessus (qui correspond en quelque sorte à une factorielle puisqu'elle contient toutes les sous-matrices possibles des tailles comprises entre 2 et k) à la puissance k, on obtient une matrice de trace k. La copie d'écran d'exécution montre la matrice initiale et la matrice élevée à la puissance k pour $k = 5$.

```python
import numpy as np
import numpy.linalg as alg
import time

for k in range(5,6):
    m=int(k*(k+1)/2-1)
    a = np.zeros((m,m))
    for u in range(k-1):
        i = int(((u+1)**2+3*(u+1)-2)/2)
        j = int(((u+3)**2-3*(u+3))/2)
        a[i,j] = 1
    for u in range(m-1):
        a[u,u+1] = 1
    for u in range(k-2):
        i = int(((u+1)**2+3*(u+1)-2)/2)
        j = int(((u+1)**2+3*(u+1)-2)/2+1)
        a[i,j] = 0
    print(a)
    grise = alg.matrix_power(a,k)
    print('')
    print('')
    print(grise)
    if (np.trace(grise) == k):
        print(k),
```



Pour terminer, la formule de calcul exact de $\pi(n)$ que j'avais proposée en septembre 2018, basée sur l'utilisation des valuations p-adiques [5].

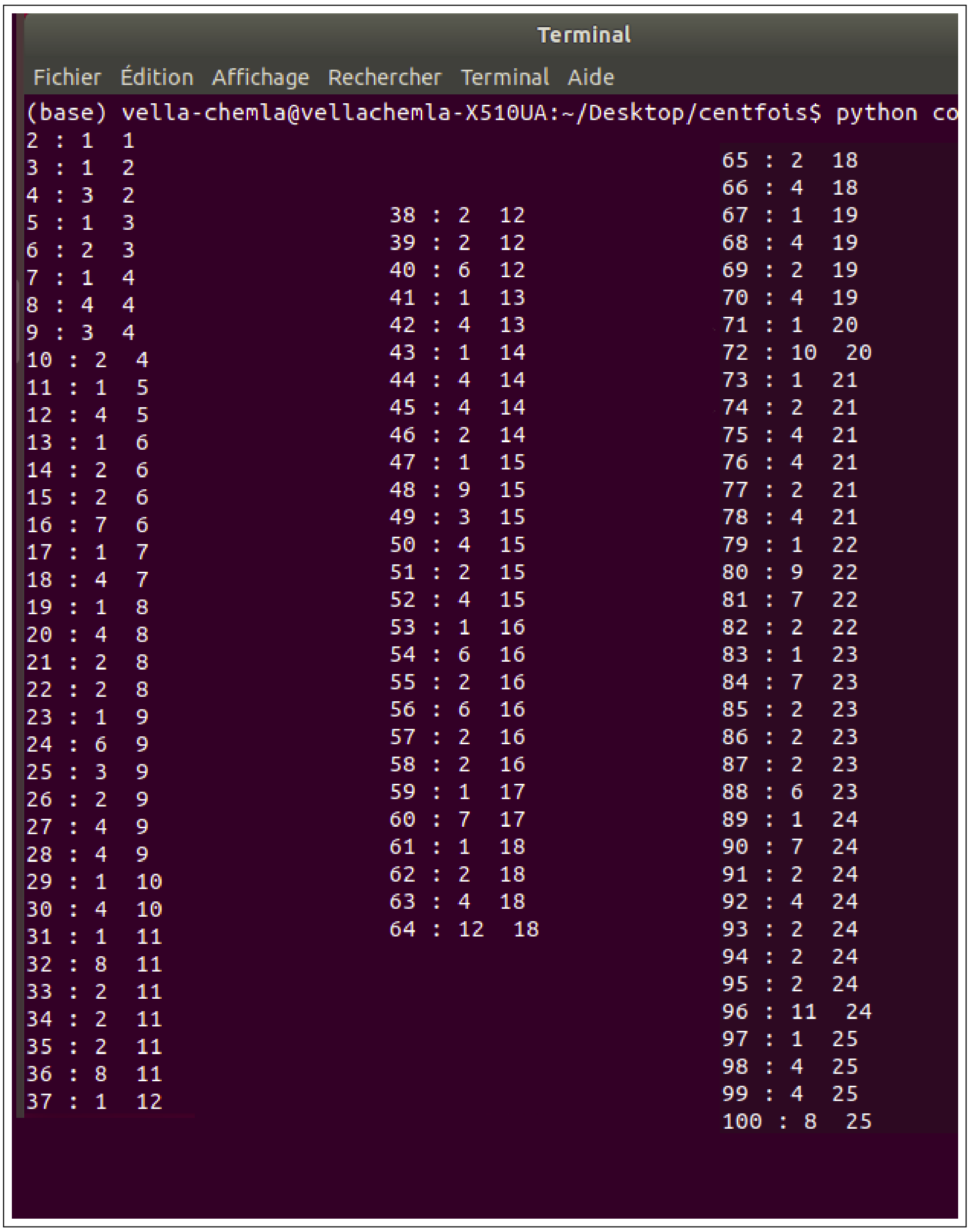
On a pour tout entier $m \geq 2$:

$$
\begin{aligned}
f(m) &= \sum_{n=2}^{\sqrt{m}} v(m,n) \\
\pi(m) &= \sum_{n=2}^{\sqrt{m}} f(m) \\
&= \sum_{n=2}^{\sqrt{m}} \left\lfloor \sum_{n=1}^{\sqrt{m}} v(m,n) \right\rfloor
\end{aligned}
$$

Voici le programme de calcul de la formule et son résultat.

```python
from math import floor, sqrt

def vp(n, p):
    if (p == 1):
        return 1
    if ((n % p) != 0):
        return 0
    else:
        return vp(n/p,p)+1

nmax = 100 ;
pix = 0 ;
for m in range(2, nmax+1):
    print(str(m)+" : ", end='')
    somme = 0
    rac = floor(sqrt(m))
    for n in range(1, rac+1):
        somme = somme+vp(m, n)
    print(str(somme)+"  ", end=''),
    pix = pix+floor(1.0/float(somme))
    print(str(pix)+"  "),
```

5. découvertes dans http://denise.vella.chemla.free.fr/fevrier2006.pdf, j'ai travaillé une première fois sur la formule proposée ici dans ces deux notes http://denise.vella.chemla.free.fr/fracto.pdf et http://denise.vella.chemla.free.fr/fractosimple.pdf.



```
Terminal
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
(base) vella-chemla@vellachemla-X510UA:~/Desktop/centfois$ python co
2 : 1  1
3 : 1  2
4 : 3  2
5 : 1  3
6 : 2  3
7 : 1  4
8 : 4  4
9 : 3  4
10 : 2  4
11 : 1  5
12 : 4  5
13 : 1  6
14 : 2  6
15 : 2  6
16 : 7  6
17 : 1  7
18 : 4  7
19 : 1  8
20 : 4  8
21 : 2  8
22 : 2  8
23 : 1  9
24 : 6  9
25 : 3  9
26 : 2  9
27 : 4  9
28 : 4  9
29 : 1  10
30 : 4  10
31 : 1  11
32 : 8  11
33 : 2  11
34 : 2  11
35 : 2  11
36 : 8  11
37 : 1  12
38 : 2  12
39 : 2  12
40 : 6  12
41 : 1  13
42 : 4  13
43 : 1  14
44 : 4  14
45 : 4  14
46 : 2  14
47 : 1  15
48 : 9  15
49 : 3  15
50 : 4  15
51 : 2  15
52 : 4  15
53 : 1  16
54 : 6  16
55 : 2  16
56 : 6  16
57 : 2  16
58 : 2  16
59 : 1  17
60 : 7  17
61 : 1  18
62 : 2  18
63 : 4  18
64 : 12  18
65 : 2  18
66 : 4  18
67 : 1  19
68 : 4  19
69 : 2  19
70 : 4  19
71 : 1  20
72 : 10  20
73 : 1  21
74 : 2  21
75 : 4  21
76 : 4  21
77 : 2  21
78 : 4  21
79 : 1  22
80 : 9  22
81 : 7  22
82 : 2  22
83 : 1  23
84 : 7  23
85 : 2  23
86 : 2  23
87 : 2  23
88 : 6  23
89 : 1  24
90 : 7  24
91 : 2  24
92 : 4  24
93 : 2  24
94 : 2  24
95 : 2  24
96 : 11  24
97 : 1  25
98 : 4  25
99 : 4  25
100 : 8  25
```

On rappelle quelques propriétés de la valuation p-adique :

- Soient m et n deux entiers ; m divise n si $v_p(m) \leq v_p(n)$ pour tout nombre premier p ;
- si a et b sont des entiers non nuls,
  $v_p(pgcd(a,b)) = min(v_p(a), v_p(b))$
  $v_p(ppcm(a,b)) = max(v_p(a), v_p(b))$;
- si a et b sont des entiers non nuls et p un nombre premier quelconque,
  $v_p(ab) = v_p(a) + v_p(b)$
  $v_p(a+b) \geq min(v_p(a), v_p(b))$.