Faire des sommes (Denise Vella-Chemla 22.1.2022)

On calcule toutes les sommes différentes de 2 nombres premiers différents que l'on peut faire à partir de l'ensemble $E = \{7, 11, 13, 17, 19, 23, 29\}$. Les voici.

$7+11=18$					
$7+13=20$	$(11+13)$				
$7+17=24$	$11+17=28$	$(13+17)$			
$7+19=26$	$(11+19)$	$13+19=32$	$(17+19)$		
$7+23=30$	$11+23=34$	$(13+23)$	$(17+23)$	$(19+23)$	
$7+29=36$	$11+29=40$	$13+29=42$	$17+29=46$	$19+29=48$	$23+29=52$

On a mis entre parenthèses les sommes égales à une somme déjà trouvée dans une colonne plus à gauche du tableau (les redondances).

Noter qu'on a "raté" les nombres pairs 22, 38, 44, 50, 52, 54 et 56, strictement compris entre 14, le double de 7, et 58, le double de 29. Les redondances posent problème : on souhaite calculer le nombre minimum de sommes différentes de 2 nombres premiers différents que l'on peut calculer.

Puisque $a < b$ et $c < d$ ont pour conséquence $a + c < b + d$, on peut, à partir des $k = 7$ nombres que contient l'ensemble E, fabriquer au moins $(k-1)+(k-2) = 2k-3$ sommes différentes de 2 nombres premiers différents suivantes :

$7+7=14$	
	$7+11=18$
$11+11=22$	
	$11+13=24$
$13+13=26$	
	$13+17=30$
$17+17=34$	
	$17+19=36$
$19+19=38$	
	$19+23=42$
$23+23=46$	
	$23+29=52$
$29+29=58$	

Entre 6 et $2p_k$, il y a $\dfrac{2p_k-4}{2} = p_k - 2$ nombres pairs différents. On a (voir [1]) $p_k \cong k\,(\log k + \log\log k - 1)$ (éno

Si on disposait de davantage de sommes différentes de 2 nombres premiers différents que de nombres pairs à calculer, on obtiendrait que la conjecture de Goldbach est vraie par le principe des tiroirs. Mais on ne dispose pas de suffisamment de sommes différentes : $2k-3 < k\left(\log k + \log\log k - \dfrac{1}{2}\right)$.

Référence

[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.

Petite note pour mémoire d'août 2020
Denise Vella-Chemla

Cette petite note est destinée à garder trace d'une expérience de programmation de la preuve par Alain Connes du théorème de Morley par un fichier texte plutôt que par le contenu d'un blog, vite enfoui par sédimentation dans les limbes de la toile.

Selon le célèbre titre de l'opus de Donald E. Knuth (*The Art Of Computer Programming*), la programmation des ordinateurs est un art. Certains informaticiens sont des artistes en effet.

La démonstration par Alain Connes du théorème de Morley est une démonstration géométrique utilisant les affixes complexes des sommets d'un triangle. On trouvera sa traduction dans l'item 47) de la liste Transcriptions AC.

Inspiré par un tableau visible à cette adresse Johnson-Smithsonian, le tableau de Crockett Johnson de 1969 intitulé CJ69, représentant le théorème de Morley, un artiste de la programmation[1] a écrit un programme esthétique en asymptote, programme dont le résultat est cette belle image jctM.jpg.

Le programme en asymptote est consultable ici : Morley-asymptote.pdf.

On peut coller[2] le code fourni juste ci-dessus à cette adresse pour l'exécuter : Asymptote en ligne.

Ici une version dynamique en geogebra : on peut modifier les positions des sommets du triangle et on peut également modifier la position du point M, laissé invariant par le produit des cubes des symétries par rapport aux côtés du triangle initial :

Théorème de Morley - démonstration d'Alain Connes en geogebra.

[1] Jacques Chemla

[2] Faire un Copier (Ctrl+C), Coller (Ctrl+V) du code dans la fenêtre de code à gauche et cliquer sur le bouton Execute, le résultat du programme apparaît dans la fenêtre de droite.

Petite note pour mémoire “Pourquoi peut-on tous les décomposer ?”
Denise Vella-Chemla
19.2.2022

Cette petite note est destinée à garder trace d’un élément trouvé en traduisant un article de Deaconescu (voir Ajouter des unités mod n).

On a vu précédemment (voir unités de plusieurs groupes à la fois) que les décomposants de Goldbach d’un nombre pair ($n = p + q$ avec p et q deux nombres premiers) sont à la fois premiers à n et premiers au produit des nombres premiers inférieurs à $\sqrt{n}$.

On se posait donc la question de savoir pourquoi il est toujours possible d’obtenir un nombre pair compris entre p_k^2 et p_{k+1}^2 (p_k et p_{k+1} deux nombres premiers successifs) par somme de deux unités qui seraient des nombres premiers à $\prod\limits_{\substack{p_j\ premier \\ p_j \leq p_k}} p_j$.

On a trouvé la réponse dans l’article de Deaconescu, dans le Corollaire 1 du Théorème, point ii) à la page 3.

Le problème est que la formule (2), permettant de démontrer le Corollaire 1 en question, est démontrée quant à elle dans la référence [1] de l’article ; or cette référence n’est pas en ligne, et on ne sait pas démontrer cette formule (2), on ne peut donc pas “disposer” du Corollaire 1 du Théorème.

Par curiosité, on a programmé le calcul (dont il est question dans l’article de Deaconescu) du nombre d’automorphismes du groupe additif ayant exactement d points fixes (noté $\Psi(d, n)$) pour voir si éventuellement, ce nombre pourrait être corrélé au nombre de décomposants de Goldbach d’un nombre pair mais sans succès.

Le résultat du programme est consultable ici Calcul du nombre d’automorphismes etc..

Unités (Denise Vella-Chemla, 14.1.2022)

On voudrait fournir ici les résultats d'un programme informatique qui montre certaines relations entre des listes de nombres, en lien avec la conjecture de Goldbach.

Le programme ci-après calcule des intersections d'ensembles de nombres, pour les nombres pairs de 6 à 100. Il s'agit de considérer, dans le résultat du programme fourni après celui-ci, la liste L3 (dont on omet le nombre 1 lorsqu'il appartient à cette liste), pour $n \geqslant 46$.

Pour les nombres pairs n compris entre 46 et 100, on constate que les nombres x qui sont à la fois premiers à n et premiers au produit des nombres premiers inférieurs à $\sqrt{n}$, que l'on notera $Prodprem$ dans la suite[1], et dont le complémentaire à n (i.e. le nombre $n-x$) est aussi premier à $Prodprem$, sont des décomposants de Goldbach de n (sauf 1, lorsque ce nombre appartient à L3). On ne sait pas démontrer que l'intersection (notée L3 ici) des listes considérées n'est jamais vide.

Voici le programme.

```
import math

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

def pgcd(m, n):
    while (m != 0):
        r = n % m
        n = m
        m = r
    return(n)

def indic(n):
    compteur = 0
    for k in range(n+1):
        if pgcd(n,k) == 1:
            compteur += 1
    return compteur
```

[1]Un nombre a est premier à un nombre b (différent de lui) si et seulement si leur pgcd est égal à 1 (i.e. si et seulement si a et b n'ont aucun facteur commun autre que 1).

```
def premiersa(n):
    Liste = []
    for k in range(n+1):
        if pgcd(n,k) == 1:
            Liste.append(k)
    return Liste

for n in range(6,102,2):
    print('n = ', n, ' -> ')
    print(' Nombres < n et premiers à n —> L1 = ', premiersa(n))
    moitie = int(n/2)
    rac = int(math.sqrt(n))
    prodprem = 1
    for x in range(rac+1):
        if prime(x):
            prodprem = prodprem * x
    liste1 = premiersa(n)
    liste2 = premiersa(prodprem)
    print(' Prodprem = produit des nombres premiers <= sqrt(n) —> ',prod-
prem)
    print(' Nombres < Prodprem et premiers à Prodprem —> L2 = ', liste2)
    liste3 = []
    for k in range(moitie):
        if k in liste1 and k in liste2 and n-k in liste2:
            liste3.append(k)
    print(' Nombres ∈ ∩(Liste1,Liste2) dont le compl. à n ∈ Liste2 —> L3 =
',liste3)
    nbsol = 0
    listedg = []
    for x in range(3,moitie+1,2):
        if prime(x) and prime(n-x):
            nbsol += 1
            listedg.append(x)
    print(' Liste des décomposants de Goldbach de n —> ', listedg)
```

Voici le résultat en sortie du programme.

n = 6 –>
Nombres < n et premiers à n —> L1 = [1, 5]
$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 2
Nombres < $Prodprem$ et premiers à $Prodprem$ —> L2 = [1]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [3]

n = 8 –>
Nombres < n et premiers à n —> L1 = [1, 3, 5, 7]
$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 2
Nombres < $Prodprem$ et premiers à $Prodprem$ —> L2 = [1]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [3]

n = 10 –>
Nombres < n et premiers à n —> L1 = [1, 3, 7, 9]
$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 6
Nombres < $Prodprem$ et premiers à $Prodprem$ —> L2 = [1, 5]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [3, 5]

n = 12 –>
Nombres < n et premiers à n —> L1 = [1, 5, 7, 11]
$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 6
Nombres < $Prodprem$ et premiers à $Prodprem$ —> L2 = [1, 5]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [5]

n = 14 –>
Nombres < n et premiers à n —> L1 = [1, 3, 5, 9, 11, 13]
$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 6
Nombres < $Prodprem$ et premiers à $Prodprem$ —> L2 = [1, 5]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [3, 7]

n = 16 –>
Nombres < n et premiers à n —> L1 = [1, 3, 5, 7, 9, 11, 13, 15]
$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 6
Nombres < $Prodprem$ et premiers à $Prodprem$ —> L2 = [1, 5]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [3, 5]

n = 18 –>
Nombres $< n$ et premiers à n —> L1 = [1, 5, 7, 11, 13, 17]
Prodprem = produit des nombres premiers $<=$ sqrt(n) —> 6
Nombres $<$ *Prodprem* et premiers à *Prodprem* —> L2 = [1, 5]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [5, 7]

n = 20 –>
Nombres $< n$ et premiers à n —> L1 = [1, 3, 7, 9, 11, 13, 17, 19]
Prodprem = produit des nombres premiers $<=$ sqrt(n) —> 6
Nombres $<$ *Prodprem* et premiers à *Prodprem* —> L2 = [1, 5]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [3, 7]

n = 22 –>
Nombres $< n$ et premiers à n —> L1 = [1, 3, 5, 7, 9, 13, 15, 17, 19, 21]
Prodprem = produit des nombres premiers $<=$ sqrt(n) —> 6
Nombres $<$ *Prodprem* et premiers à *Prodprem* —> L2 = [1, 5]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [3, 5, 11]

n = 24 –>
Nombres $< n$ et premiers à n —> L1 = [1, 5, 7, 11, 13, 17, 19, 23]
Prodprem = produit des nombres premiers $<=$ sqrt(n) —> 6
Nombres $<$ *Prodprem* et premiers à *Prodprem* —> L2 = [1, 5]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [5, 7, 11]

n = 26 –>
Nombres $< n$ et premiers à n —> L1 = [1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25]
Prodprem = produit des nombres premiers $<=$ sqrt(n) —> 30
Nombres $<$ *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = [7]
Liste des décomposants de Goldbach de n —> [3, 7, 13]

n = 28 –>
Nombres $< n$ et premiers à n —> L1 = [1, 3, 5, 9, 11, 13, 15, 17, 19, 23, 25, 27]
Prodprem = produit des nombres premiers $<=$ sqrt(n) —> 30
Nombres $<$ *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]
Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = [11]
Liste des décomposants de Goldbach de n —> [5, 11]

$n\ =\ 30$ –>

Nombres $< n$ et premiers à n —> L1 = [1, 7, 11, 13, 17, 19, 23, 29]

Prodprem = produit des nombres premiers <= sqrt(n) —> 30

Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]

Nombres $\in\ \cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [1, 7, 11, 13]

Liste des décomposants de Goldbach de n —> [7, 11, 13]

$n\ =\ 32$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31]

Prodprem = produit des nombres premiers <= sqrt(n) —> 30

Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]

Nombres $\in\ \cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = [13]

Liste des décomposants de Goldbach de n —> [3, 13]

$n\ =\ 34$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 19, 21, 23, 25, 27, 29, 31, 33]

Prodprem = produit des nombres premiers <= sqrt(n) —> 30

Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]

Nombres $\in\ \cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = [11]

Liste des décomposants de Goldbach de n —> [3, 5, 11, 17]

$n\ =\ 36$ –>

Nombres $< n$ et premiers à n —> L1 = [1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, 35]

Prodprem = produit des nombres premiers <= sqrt(n) —> 30

Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]

Nombres $\in\ \cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [7, 13, 17]

Liste des décomposants de Goldbach de n —> [5, 7, 13, 17]

$n\ =\ 38$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 21, 23, 25, 27, 29, 31, 33, 35, 37]

Prodprem = produit des nombres premiers <= sqrt(n) —> 30

Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]

Nombres $\in\ \cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = []

Liste des décomposants de Goldbach de n —> [7, 19]

n = 40 –>
Nombres < n et premiers à n —>
L1 = [1, 3, 7, 9, 11, 13, 17, 19, 21, 23, 27, 29, 31, 33, 37, 39]
Prodprem = produit des nombres premiers <= sqrt(n) —> 30
Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]
Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —> L3 = [11, 17]
Liste des décomposants de Goldbach de n —> [3, 11, 17]

n = 42 –>
Nombres < n et premiers à n —> L1 = [1, 5, 11, 13, 17, 19, 23, 25, 29, 31, 37, 41]
Prodprem = produit des nombres premiers <= sqrt(n) —> 30
Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]
Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —> L3 = [13, 19]
Liste des décomposants de Goldbach de n —> [5, 11, 13, 19]

n = 44 –>
Nombres < n et premiers à n —>
L1 = [1, 3, 5, 7, 9, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 35, 37, 39, 41, 43]
Prodprem = produit des nombres premiers <= sqrt(n) —> 30
Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]
Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —> L3 = []
Liste des décomposants de Goldbach de n —> [3, 7, 13]

n = 46 –>
Nombres < n et premiers à n —>
L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45]
Prodprem = produit des nombres premiers <= sqrt(n) —> 30
Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]
Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —> L3 = [17]
Liste des décomposants de Goldbach de n —> [3, 5, 17, 23]

n = 48 –>
Nombres < n et premiers à n —>
L1 = [1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47]
Prodprem = produit des nombres premiers <= sqrt(n) —> 30
Nombres < *Prodprem* et premiers à *Prodprem* —> L2 = [1, 7, 11, 13, 17, 19, 23, 29]
Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —> L3 = [19]
Liste des décomposants de Goldbach de n —> [5, 7, 11, 17, 19]

n = 50 –>

Nombres < n et premiers à n —>

L1 = [1, 3, 7, 9, 11, 13, 17, 19, 21, 23, 27, 29, 31, 33, 37, 39, 41, 43, 47, 49]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —> L3 = [13, 19]

Liste des décomposants de Goldbach de n —> [3, 7, 13, 19]

n = 52 –>

Nombres < n et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 41, 43, 45, 47, 49, 51]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —> L3 = [11, 23]

Liste des décomposants de Goldbach de n —> [5, 11, 23]

n = 54 –>

Nombres < n et premiers à n —>

L1 = [1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47, 49, 53]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —>

L3 = [1, 11, 13, 17, 23]

Liste des décomposants de Goldbach de n —> [7, 11, 13, 17, 23]

n = 56 –>

Nombres < n et premiers à n —>

L1 = [1, 3, 5, 9, 11, 13, 15, 17, 19, 23, 25, 27, 29, 31, 33, 37, 39, 41, 43, 45, 47, 51, 53, 55]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —> L3 = [13, 19]

Liste des décomposants de Goldbach de n —> [3, 13, 19]

n = 58 –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à $n \in$ L2 —> L3 = [11, 17]

Liste des décomposants de Goldbach de n —> [5, 11, 17, 29]

n = 60 –>

Nombres $< n$ et premiers à n —>

L1 = [1, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 49, 53, 59]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [1, 13, 17, 19, 23, 29]

Liste des décomposants de Goldbach de n —> [7, 13, 17, 19, 23, 29]

n = 62 –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à $n \in$ L2 —> L3 = [1, 19]

Liste des décomposants de Goldbach de n —> [3, 19, 31]

n = 64 –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [11, 17, 23]

Liste des décomposants de Goldbach de n —> [3, 5, 11, 17, 23]

$n = 66$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 5, 7, 13, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47, 49, 53, 59, 61, 65]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = [13, 19, 23, 29]

Liste des décomposants de Goldbach de n —> [5, 7, 13, 19, 23, 29]

$n = 68$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 53, 55, 57, 59, 61, 63, 65, 67]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —> L3 = [1, 31]

Liste des décomposants de Goldbach de n —> [7, 31]

$n = 70$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 9, 11, 13, 17, 19, 23, 27, 29, 31, 33, 37, 39, 41, 43, 47, 51, 53, 57, 59, 61, 67, 69]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [11, 17, 23, 29]

Liste des décomposants de Goldbach de n —> [3, 11, 17, 23, 29]

$n = 72$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47, 49, 53, 55, 59, 61, 65, 67, 71]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in \cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [1, 11, 13, 19, 29, 31]

Liste des décomposants de Goldbach de n —> [5, 11, 13, 19, 29, 31]

$n = 74$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in \cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [1, 13, 31]

Liste des décomposants de Goldbach de n —> [3, 7, 13, 31, 37]

$n = 76$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 59, 61, 63, 65, 67, 69, 71, 73, 75]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in \cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [17, 23, 29]

Liste des décomposants de Goldbach de n —> [3, 5, 17, 23, 29]

$n = 78$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 5, 7, 11, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47, 49, 53, 55, 59, 61, 67, 71, 73, 77]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in \cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [11, 17, 19, 31, 37]

Liste des décomposants de Goldbach de n —> [5, 7, 11, 17, 19, 31, 37]

$n = 80$ –> Nombres $< n$ et premiers à n —>

L1 = [1, 3, 7, 9, 11, 13, 17, 19, 21, 23, 27, 29, 31, 33, 37, 39, 41, 43, 47, 49, 51, 53, 57, 59, 61, 63, 67, 69, 71, 73, 77, 79]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in \cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [1, 13, 19, 37]

Liste des décomposants de Goldbach de n —> [7, 13, 19, 37]

$n = 82$ –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in \cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [11, 23, 29]

Liste des décomposants de Goldbach de n —> [3, 11, 23, 29, 41]

n = 84 –>

Nombres $< n$ et premiers à n —>

L1 = [1, 5, 11, 13, 17, 19, 23, 25, 29, 31, 37, 41, 43, 47, 53, 55, 59, 61, 65, 67, 71, 73, 79, 83]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [1, 11, 13, 17, 23, 31, 37, 41]

Liste des décomposants de Goldbach de n —> [5, 11, 13, 17, 23, 31, 37, 41]

n = 86 –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [13, 19]

Liste des décomposants de Goldbach de n —> [3, 7, 13, 19, 43]

n = 88 –>

Nombres $< n$ et premiers à n —>

L1 = [1, 3, 5, 7, 9, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 79, 81, 83, 85, 87]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [17, 29, 41]

Liste des décomposants de Goldbach de n —> [5, 17, 29, 41]

n = 90 –>

Nombres < n et premiers à n —>

L1 = [1, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 49, 53, 59, 61, 67, 71, 73, 77, 79, 83, 89]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [1, 11, 17, 19, 23, 29, 31, 37, 43]

Liste des décomposants de Goldbach de n —> [7, 11, 17, 19, 23, 29, 31, 37, 43]

n = 92 –>

Nombres < n et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [13, 19, 31]

Liste des décomposants de Goldbach de n —> [3, 13, 19, 31]

n = 94 –>

Nombres < n et premiers à n —>

L1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93]

$Prodprem$ = produit des nombres premiers <= sqrt(n) —> 210

Nombres < $Prodprem$ et premiers à $Prodprem$ —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres $\in$ $\cap$(L1,L2) dont le compl. à $n \in$ L2 —>

L3 = [11, 23, 41]

Liste des décomposants de Goldbach de n —> [5, 11, 23, 41, 47]

n = 96 –>

Nombres < n et premiers à n —>

L1 = [1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47, 49, 53, 55, 59, 61, 65, 67, 71, 73, 77, 79, 83, 85, 89, 91, 95]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —>

L3 = [13, 17, 23, 29, 37, 43]

Liste des décomposants de Goldbach de n —> [7, 13, 17, 23, 29, 37, 43]

n = 98 –>

Nombres < n et premiers à n —>

L1 = [1, 3, 5, 9, 11, 13, 15, 17, 19, 23, 25, 27, 29, 31, 33, 37, 39, 41, 43, 45, 47, 51, 53, 55, 57, 59, 61, 65, 67, 69, 71, 73, 75, 79, 81, 83, 85, 87, 89, 93, 95, 97]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ L2 —>

L3 = [1, 19, 31, 37]

Liste des décomposants de Goldbach de n —> [19, 31, 37]

n = 100 –>

Nombres < n et premiers à n —>

L1 = [1, 3, 7, 9, 11, 13, 17, 19, 21, 23, 27, 29, 31, 33, 37, 39, 41, 43, 47, 49, 51, 53, 57, 59, 61, 63, 67, 69, 71, 73, 77, 79, 81, 83, 87, 89, 91, 93, 97, 99]

Prodprem = produit des nombres premiers <= sqrt(n) —> 210

Nombres < *Prodprem* et premiers à *Prodprem* —>

L2 = [1, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 121, 127, 131, 137, 139, 143, 149, 151, 157, 163, 167, 169, 173, 179, 181, 187, 191, 193, 197, 199, 209]

Nombres ∈ ∩(L1,L2) dont le compl. à n ∈ ∩ L2 —>

L3 = [11, 17, 29, 41, 47]

Liste des décomposants de Goldbach de n —> [3, 11, 17, 29, 41, 47]

Tout x qui est une unité à n (un nombre premier à n) vérifie l'équation modulaire

$$x^{\varphi(n)} \equiv 1 \pmod{n}$$

On vérifie par exemple, puisque $\varphi(98) = 42$ et $\varphi(210) = 48$, que 19, décomposant de Goldbach de

98, vérifie le système de trois équations modulaires :

$$\begin{cases} 19^{42} \equiv 1 \pmod{98} \\ 19^{48} \equiv 1 \pmod{210} \\ 79^{48} \equiv 1 \pmod{210} \end{cases}$$

tandis que 17 n'est pas un décomposant de Goldbach de 98 parce que $98 - 17 = 81$ et $81^{48} \not\equiv 1 \pmod{210}$.

Un décomposant de Goldbach x du nombre pair n doit donc rendre vraies les 3 équations modulaires suivantes

$$\begin{cases} x^{\varphi(n)} & \equiv 1 \pmod{n} \\ x^{\varphi(Prodprem)} & \equiv 1 \pmod{Prodprem} \\ (n-x)^{\varphi(Prodprem)} & \equiv 1 \pmod{Prodprem} \end{cases}$$

avec

$$\varphi(Propdprem) = \prod_{\substack{p \text{ premier} \\ 3 \leqslant p \leqslant \sqrt{n}}} (p-1)$$

et

$$(n-x)^{\varphi(Prodprem)} = \sum_{k=0}^{\varphi(Prodprem)} \begin{pmatrix} \varphi(Prodprem) \\ k \end{pmatrix} n^{\varphi(Prodprem)-k}(-x)^k$$

La notation $\begin{pmatrix} \varphi(Prodprem) \\ k \end{pmatrix}$ représente le coefficient binomial qui intervient dans le calcul du développement de la puissance de $(n - x)$.

Il faudrait comprendre pourquoi un tel système de trois équations modulaires a toujours une solution comprise entre 3 et $n/2$.

On modifie le programme en conséquence.

```
import math

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

def pgcd(m, n):
        while (m != 0):
                r = n % m
                n = m
                m = r
        return(n)
```

```
def indic(n):
        phi = 0
        for k in range(n):
                if pgcd(n,k) == 1:
                        phi = phi+1
        print(' ',n, 'indic ', phi)
        return phi
def premiersa(n):
        liste = []
        for k in range(n+1):
                if pgcd(n,k) == 1:
                        liste.append(k)
        return liste

def puissance(x, k, n):
        expo = 1
        puiss = 1
        while expo ¡= k:
                puiss = (puiss*x)%n
                expo = expo+1
        print(x, '**', k, '=', puiss, '(mod ',n,')')
        return puiss

for n in range(6,102,2):
        print('=', n, ' –> ')
        moitie = int(n/2)
        rac = int(math.sqrt(n))
        prodprem = 1
        for x in range(rac+1):
                if prime(x):
                        prodprem = prodprem * x
        varphi1 = indic(n)
        varphi2 = indic(prodprem)
        listexverifiantlestroisegalites = []
        for x in range(moitie):
                if puissance(x,varphi1,n) == 1 and puissance(x,varphi2,prodprem) ==
1 and puissance(n-x,varphi2,prodprem) == 1:
                        print(' ',x, 'verifie les 3 egalites.')
                        listexverifiantlestroisegalites.append(x)
        print(' ',listexverifiantlestroisegalites)
        nbsol = 0
        listedg = []
        for x in range(3,moitie+1,2):
                if prime(x) and prime(n-x):
                        nbsol += 1
                        listedg.append(x)
        print(' Liste des decomposants de Goldbach de ',n,' —> ', listedg)
        print(' card = ', nbsol)
```

Voici le résultat du second programme.

```
n = 6 –>
    6 indic 2
    2 indic 1
    1 verifie les 3 egalites.
    [1]
    Liste des decomposants de Goldbach de 6 —> [3]
    card = 1

n = 8 –>
    8 indic 4
    2 indic 1
    1 verifie les 3 egalites.
    3 verifie les 3 egalites.
    [1, 3]
    Liste des decomposants de Goldbach de 8 —> [3]
    card = 1

n = 10 –>
    10 indic 4
    6 indic 2
    []
    Liste des decomposants de Goldbach de 10 —> [3, 5]
    card = 2

n = 12 –>
    12 indic 4
    6 indic 2
    1 verifie les 3 egalites.
    5 verifie les 3 egalites.
    [1, 5]
    Liste des decomposants de Goldbach de 12 —> [5]
    card = 1

n = 14 –>
    14 indic 6
    6 indic 2
    1 verifie les 3 egalites.
    [1]
    Liste des decomposants de Goldbach de 14 —> [3, 7]
    card = 2
```

```
n = 16 –>
        16 indic 8
        6 indic 2
        5 verifie les 3 egalites.
        [5]
        Liste des decomposants de Goldbach de 16 —> [3, 5]
        card = 2

n = 18 –>
        18 indic 6
        6 indic 2
        1 verifie les 3 egalites.
        5 verifie les 3 egalites.
        7 verifie les 3 egalites.
        [1, 5, 7]
        Liste des decomposants de Goldbach de 18 —> [5, 7]
        card = 2

n = 20 –>
        20 indic 8
        6 indic 2
        1 verifie les 3 egalites.
        7 verifie les 3 egalites.
        [1, 7]
        Liste des decomposants de Goldbach de 20 —> [3, 7]
        card = 2

n = 22 –>
        22 indic 10
        6 indic 2
        5 verifie les 3 egalites.
        [5]
        Liste des decomposants de Goldbach de 22 —> [3, 5, 11]
        card = 3

n = 24 –>
        24 indic 8
        6 indic 2
        1 verifie les 3 egalites.
        5 verifie les 3 egalites.
        7 verifie les 3 egalites.
        11 verifie les 3 egalites.
        [1, 5, 7, 11]
        Liste des decomposants de Goldbach de 24 —> [5, 7, 11]
        card = 3
```

```
n = 26 –>
    26 indic 12
    30 indic 8
    7 verifie les 3 egalites.
    [7]
    Liste des decomposants de Goldbach de 26 —> [3, 7, 13]
    card = 3

n = 28 –>
    28 indic 12
    30 indic 8
    11 verifie les 3 egalites.
    [11]
    Liste des decomposants de Goldbach de 28 —> [5, 11]
    card = 2

n = 30 –>
    30 indic 8
    30 indic 8
    1 verifie les 3 egalites.
    7 verifie les 3 egalites.
    11 verifie les 3 egalites.
    13 verifie les 3 egalites.
    [1, 7, 11, 13]
    Liste des decomposants de Goldbach de 30 —> [7, 11, 13]
    card = 3

n = 32 –>
    32 indic 16
    30 indic 8
    1 verifie les 3 egalites.
    13 verifie les 3 egalites.
    [1, 13]
    Liste des decomposants de Goldbach de 32 —> [3, 13]
    card = 2

n = 34 –>
    34 indic 16
    30 indic 8
    11 verifie les 3 egalites.
    [11]
    Liste des decomposants de Goldbach de 34 —> [3, 5, 11, 17]
    card = 4
```

```
n = 36 ->
    36 indic 12
    30 indic 8
    7 verifie les 3 egalites.
    13 verifie les 3 egalites.
    17 verifie les 3 egalites.
    [7, 13, 17]
    Liste des decomposants de Goldbach de 36 —> [5, 7, 13, 17]
    card = 4

n = 38 ->
    38 indic 18
    30 indic 8
    1 verifie les 3 egalites.
    7 verifie les 3 egalites.
    [1, 7]
    Liste des decomposants de Goldbach de 38 —> [7, 19]
    card = 2

n = 40 ->
    40 indic 16
    30 indic 8
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    [11, 17]
    Liste des decomposants de Goldbach de 40 —> [3, 11, 17]
    card = 3

n = 42 ->
    42 indic 12
    30 indic 8
    1 verifie les 3 egalites.
    11 verifie les 3 egalites.
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [1, 11, 13, 19]
    Liste des decomposants de Goldbach de 42 —> [5, 11, 13, 19]
    card = 4

n = 44 ->
    44 indic 20
    30 indic 8
    1 verifie les 3 egalites.
    7 verifie les 3 egalites.
    13 verifie les 3 egalites.
    [1, 7, 13]
    Liste des decomposants de Goldbach de 44 —> [3, 7, 13]
    card = 3
```

```
n = 46 ->
    46 indic 22
    30 indic 8
    17 verifie les 3 egalites.
    [17]
    Liste des decomposants de Goldbach de 46 —> [3, 5, 17, 23]
    card = 4

n = 48 ->
    48 indic 16
    30 indic 8
    1 verifie les 3 egalites.
    7 verifie les 3 egalites.
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [1, 7, 11, 17, 19]
    Liste des decomposants de Goldbach de 48 —> [5, 7, 11, 17, 19]
    card = 5

n = 50 ->
    50 indic 20
    210 indic 48
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [13, 19]
    Liste des decomposants de Goldbach de 50 —> [3, 7, 13, 19]
    card = 4

n = 52 ->
    52 indic 24
    210 indic 48
    11 verifie les 3 egalites.
    23 verifie les 3 egalites.
    [11, 23]
    Liste des decomposants de Goldbach de 52 —> [5, 11, 23]
    card = 3
```

```
n = 54 –>
    54 indic 18
    210 indic 48
    1 verifie les 3 egalites.
    11 verifie les 3 egalites.
    13 verifie les 3 egalites.
    17 verifie les 3 egalites.
    23 verifie les 3 egalites.
    [1, 11, 13, 17, 23]
    Liste des decomposants de Goldbach de 54 —> [7, 11, 13, 17, 23]
    card = 5

n = 56 –>
    56 indic 24
    210 indic 48
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [13, 19]
    Liste des decomposants de Goldbach de 56 —> [3, 13, 19]
    card = 3

n = 58 –>
    58 indic 28
    210 indic 48
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    [11, 17]
    Liste des decomposants de Goldbach de 58 —> [5, 11, 17, 29]
    card = 4

n = 60 –>
    60 indic 16
    210 indic 48
    1 verifie les 3 egalites.
    13 verifie les 3 egalites.
    17 verifie les 3 egalites.
    19 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    [1, 13, 17, 19, 23, 29]
    Liste des decomposants de Goldbach de 60 —> [7, 13, 17, 19, 23, 29]
    card = 6
```

```
n = 62 –>
    62 indic 30
    210 indic 48
    1 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [1, 19]
    Liste des decomposants de Goldbach de 62 —> [3, 19, 31]
    card = 3

n = 64 –>
    64 indic 32
    210 indic 48
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    23 verifie les 3 egalites.
    [11, 17, 23]
    Liste des decomposants de Goldbach de 64 —> [3, 5, 11, 17, 23]
    card = 5

n = 66 –>
    66 indic 20
    210 indic 48
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    [13, 19, 23, 29]
    Liste des decomposants de Goldbach de 66 —> [5, 7, 13, 19, 23, 29]
    card = 6

n = 68 –>
    68 indic 32
    210 indic 48
    1 verifie les 3 egalites.
    31 verifie les 3 egalites.
    [1, 31]
    Liste des decomposants de Goldbach de 68 —> [7, 31]
    card = 2
```

```
n = 70 ->
    70 indic 24
    210 indic 48
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    [11, 17, 23, 29]
    Liste des decomposants de Goldbach de 70 —> [3, 11, 17, 23, 29]
    card = 5

n = 72 ->
    72 indic 24
    210 indic 48
    1 verifie les 3 egalites.
    11 verifie les 3 egalites.
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    29 verifie les 3 egalites.
    31 verifie les 3 egalites.
    [1, 11, 13, 19, 29, 31]
    Liste des decomposants de Goldbach de 72 —> [5, 11, 13, 19, 29, 31]
    card = 6

n = 74 ->
    74 indic 36
    210 indic 48
    1 verifie les 3 egalites.
    13 verifie les 3 egalites.
    31 verifie les 3 egalites.
    [1, 13, 31]
    Liste des decomposants de Goldbach de 74 —> [3, 7, 13, 31, 37]
    card = 5

n = 76 ->
    76 indic 36
    210 indic 48
    17 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    [17, 23, 29]
    Liste des decomposants de Goldbach de 76 —> [3, 5, 17, 23, 29]
    card = 5
```

n = 78 –>
78 indic 24
210 indic 48
11 verifie les 3 egalites.
17 verifie les 3 egalites.
19 verifie les 3 egalites.
31 verifie les 3 egalites.
37 verifie les 3 egalites.
[11, 17, 19, 31, 37]
Liste des decomposants de Goldbach de 78 —> [5, 7, 11, 17, 19, 31, 37]
card = 7

n = 80 –>
80 indic 32
210 indic 48
1 verifie les 3 egalites.
13 verifie les 3 egalites.
19 verifie les 3 egalites.
37 verifie les 3 egalites.
[1, 13, 19, 37]
Liste des decomposants de Goldbach de 80 —> [7, 13, 19, 37]
card = 4

n = 82 –>
82 indic 40
210 indic 48
11 verifie les 3 egalites.
23 verifie les 3 egalites.
29 verifie les 3 egalites.
[11, 23, 29]
Liste des decomposants de Goldbach de 82 —> [3, 11, 23, 29, 41]
card = 5

n = 84 –>
84 indic 24
210 indic 48
1 verifie les 3 egalites.
11 verifie les 3 egalites.
13 verifie les 3 egalites.
17 verifie les 3 egalites.
23 verifie les 3 egalites.
31 verifie les 3 egalites.
37 verifie les 3 egalites.
41 verifie les 3 egalites.
[1, 11, 13, 17, 23, 31, 37, 41]
Liste des decomposants de Goldbach de 84 —> [5, 11, 13, 17, 23, 31, 37, 41]
card = 8

```
n = 86 ->
	86 indic 42
	210 indic 48
	13 verifie les 3 egalites.
	19 verifie les 3 egalites.
	[13, 19]
	Liste des decomposants de Goldbach de 86 —> [3, 7, 13, 19, 43]
	card = 5

n = 88 ->
	88 indic 40
	210 indic 48
	17 verifie les 3 egalites.
	29 verifie les 3 egalites.
	41 verifie les 3 egalites.
	[17, 29, 41]
	Liste des decomposants de Goldbach de 88 —> [5, 17, 29, 41]
	card = 4

n = 90 ->
	90 indic 24
	210 indic 48
	1 verifie les 3 egalites.
	11 verifie les 3 egalites.
	17 verifie les 3 egalites.
	19 verifie les 3 egalites.
	23 verifie les 3 egalites.
	29 verifie les 3 egalites.
	31 verifie les 3 egalites.
	37 verifie les 3 egalites.
	43 verifie les 3 egalites.
	[1, 11, 17, 19, 23, 29, 31, 37, 43]
	Liste des decomposants de Goldbach de 90 —> [7, 11, 17, 19, 23, 29, 31, 37,
43]
	card = 9

n = 92 ->
	92 indic 44
	210 indic 48
	13 verifie les 3 egalites.
	19 verifie les 3 egalites.
	31 verifie les 3 egalites.
	[13, 19, 31]
	Liste des decomposants de Goldbach de 92 —> [3, 13, 19, 31]
	card = 4
```

n = 94 –>

94 indic 46
210 indic 48
11 verifie les 3 egalites.
23 verifie les 3 egalites.
41 verifie les 3 egalites.
[11, 23, 41]
Liste des decomposants de Goldbach de 94 —> [5, 11, 23, 41, 47]
card = 5

n = 96 –>

96 indic 32
210 indic 48
13 verifie les 3 egalites.
17 verifie les 3 egalites.
23 verifie les 3 egalites.
29 verifie les 3 egalites.
37 verifie les 3 egalites.
43 verifie les 3 egalites.
[13, 17, 23, 29, 37, 43]
Liste des decomposants de Goldbach de 96 —> [7, 13, 17, 23, 29, 37, 43]
card = 7

n = 98 –>

98 indic 42
210 indic 48
1 verifie les 3 egalites.
19 verifie les 3 egalites.
31 verifie les 3 egalites.
37 verifie les 3 egalites.
[1, 19, 31, 37]
Liste des decomposants de Goldbach de 98 —> [19, 31, 37]
card = 3

n = 100 –>

100 indic 40
210 indic 48
11 verifie les 3 egalites.
17 verifie les 3 egalites.
29 verifie les 3 egalites.
41 verifie les 3 egalites.
47 verifie les 3 egalites.
[11, 17, 29, 41, 47]
Liste des decomposants de Goldbach de 100 —> [3, 11, 17, 29, 41, 47]
card = 6

Unités (Denise Vella-Chemla, 14.1.2022)

On voudrait fournir ici les conclusions obtenues à partir de l'exécution d'un programme informatique qui montre certaines relations entre des listes de nombres, en lien avec la conjecture de Goldbach.

Le programme calcule des intersections d'ensembles de nombres, pour les nombres pairs de 6 à 100. Il s'agit de considérer la liste de nombres intersection de deux listes de nombres (on omet le nombre 1 lorsqu'il appartient à l'intersection de listes en question), pour $n \geqslant 46$.

Pour les nombres pairs n compris entre 46 et 100, on constate que les nombres x qui sont à la fois premiers à n et premiers au produit des nombres premiers inférieurs à $\sqrt{n}$, que l'on notera *Prodprem* dans la suite[1], et dont le complémentaire à n (i.e. le nombre $n-x$) est aussi premier à *Prodprem*, sont des décomposants de Goldbach de n (sauf 1, lorsque ce nombre appartient à L3). On ne sait pas démontrer que l'intersection (notée L3 ici) des listes considérées n'est jamais vide.

Tout x qui est une unité à n (un nombre premier à n) vérifie l'équation modulaire

$$x^{\varphi(n)} \equiv 1 \pmod{n}$$

On vérifie par exemple, puisque $\varphi(98) = 42$ et $\varphi(210) = 48$, que 19, décomposant de Goldbach de 98, vérifie le système de trois équations modulaires :

$$\begin{cases} 19^{42} \equiv 1 \pmod{98} \\ 19^{48} \equiv 1 \pmod{210} \\ 79^{48} \equiv 1 \pmod{210} \end{cases}$$

tandis que 17 n'est pas un décomposant de Goldbach de 98 parce que $17^{48} \not\equiv 1 \pmod{210}$.

Un décomposant de Goldbach x du nombre pair n doit donc rendre vraies les 3 équations modulaires suivantes

$$\begin{cases} x^{\varphi(n)} & \equiv 1 \pmod{n} \\ x^{\varphi(Prodprem)} & \equiv 1 \pmod{Prodprem} \\ (n-x)^{\varphi(Prodprem)} & \equiv 1 \pmod{Prodprem} \end{cases}$$

avec

$$\varphi(Propdprem) = \prod_{\substack{p \text{ premier} \\ 3 \leqslant p \leqslant \sqrt{n}}} (p-1)$$

et

$$(n-x)^{\varphi(Prodprem)} = \sum_{k=0}^{\varphi(Prodprem)} \left(\begin{array}{c} \varphi(Prodprem) \\ k \end{array} \right) n^{\varphi(Prodprem)-k} (-x)^k$$

La notation $\left(\begin{array}{c} \varphi(Prodprem) \\ k \end{array} \right)$ représente le coefficient binomial qui intervient dans le calcul du développement de la puissance de $(n-x)$.

[1] Un nombre a est premier à un nombre b (différent de lui) si et seulement si leur pgcd est égal à 1 (i.e. si et seulement si a et b n'ont aucun facteur commun autre que 1).

Il faudrait comprendre pourquoi un tel système de trois équations modulaires a toujours une solution comprise entre 3 et $n/2$.

On écrit le programme suivant pour vérifier notre découverte.

```
import math

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

def pgcd(m, n):
    while (m != 0):
        r = n % m
        n = m
        m = r
    return(n)

def indic(n):
    phi = 0
    for k in range(n):
        if pgcd(n,k) == 1:
            phi = phi+1
    print(' ',n, 'indic ', phi)
    return phi
def premiersa(n):
    liste = []
    for k in range(n+1):
        if pgcd(n,k) == 1:
            liste.append(k)
    return liste
```

```
def puissance(x, k, n):
    expo = 1
    puiss = 1
    while expo ¡= k:
        puiss = (puiss*x)%n
        expo = expo+1
    print(x, ’**’, k, ’=’, puiss, ’(mod ’,n,’)’)
    return puiss

for n in range(6,102,2):
    print(’=’, n, ’ –> ’)
    moitie = int(n/2)
    rac = int(math.sqrt(n))
    prodprem = 1
    for x in range(rac+1):
        if prime(x):
            prodprem = prodprem * x
    varphi1 = indic(n)
    varphi2 = indic(prodprem)
    listexverifiantlestroisegalites = []
    for x in range(moitie):
        if puissance(x,varphi1,n) == 1 and puissance(x,varphi2,prodprem) ==
1 and puissance(n-x,varphi2,prodprem) == 1:
            print(’ ’,x, ’verifie les 3 egalites.’)
            listexverifiantlestroisegalites.append(x)
    print(’ ’,listexverifiantlestroisegalites)
    nbsol = 0
    listedg = []
    for x in range(3,moitie+1,2):
        if prime(x) and prime(n-x):
            nbsol += 1
            listedg.append(x)
    print(’ Liste des decomposants de Goldbach de ’,n,’ —> ’, listedg)
    print(’ card = ’, nbsol)
```

Voici le résultat du programme ci-dessus.

```
n = 6 –>
        6 indic 2
        2 indic 1
        1 verifie les 3 egalites.
        [1]
        Liste des decomposants de Goldbach de 6 —> [3]
        card = 1

n = 8 –>
        8 indic 4
        2 indic 1
        1 verifie les 3 egalites.
        3 verifie les 3 egalites.
        [1, 3]
        Liste des decomposants de Goldbach de 8 —> [3]
        card = 1

n = 10 –>
        10 indic 4
        6 indic 2
        []
        Liste des decomposants de Goldbach de 10 —> [3, 5]
        card = 2

n = 12 –>
        12 indic 4
        6 indic 2
        1 verifie les 3 egalites.
        5 verifie les 3 egalites.
        [1, 5]
        Liste des decomposants de Goldbach de 12 —> [5]
        card = 1

n = 14 –>
        14 indic 6
        6 indic 2
        1 verifie les 3 egalites.
        [1]
        Liste des decomposants de Goldbach de 14 —> [3, 7]
        card = 2
```

```
n = 16 –>
    16 indic 8
    6 indic 2
    5 verifie les 3 egalites.
    [5]
    Liste des decomposants de Goldbach de 16 —> [3, 5]
    card = 2

n = 18 –>
    18 indic 6
    6 indic 2
    1 verifie les 3 egalites.
    5 verifie les 3 egalites.
    7 verifie les 3 egalites.
    [1, 5, 7]
    Liste des decomposants de Goldbach de 18 —> [5, 7]
    card = 2

n = 20 –>
    20 indic 8
    6 indic 2
    1 verifie les 3 egalites.
    7 verifie les 3 egalites.
    [1, 7]
    Liste des decomposants de Goldbach de 20 —> [3, 7]
    card = 2

n = 22 –>
    22 indic 10
    6 indic 2
    5 verifie les 3 egalites.
    [5]
    Liste des decomposants de Goldbach de 22 —> [3, 5, 11]
    card = 3

n = 24 –>
    24 indic 8
    6 indic 2
    1 verifie les 3 egalites.
    5 verifie les 3 egalites.
    7 verifie les 3 egalites.
    11 verifie les 3 egalites.
    [1, 5, 7, 11]
    Liste des decomposants de Goldbach de 24 —> [5, 7, 11]
    card = 3
```

```
n = 26 ->
    26 indic 12
    30 indic 8
    7 verifie les 3 egalites.
    [7]
    Liste des decomposants de Goldbach de 26 —> [3, 7, 13]
    card = 3

n = 28 ->
    28 indic 12
    30 indic 8
    11 verifie les 3 egalites.
    [11]
    Liste des decomposants de Goldbach de 28 —> [5, 11]
    card = 2

n = 30 ->
    30 indic 8
    30 indic 8
    1 verifie les 3 egalites.
    7 verifie les 3 egalites.
    11 verifie les 3 egalites.
    13 verifie les 3 egalites.
    [1, 7, 11, 13]
    Liste des decomposants de Goldbach de 30 —> [7, 11, 13]
    card = 3

n = 32 ->
    32 indic 16
    30 indic 8
    1 verifie les 3 egalites.
    13 verifie les 3 egalites.
    [1, 13]
    Liste des decomposants de Goldbach de 32 —> [3, 13]
    card = 2

n = 34 ->
    34 indic 16
    30 indic 8
    11 verifie les 3 egalites.
    [11]
    Liste des decomposants de Goldbach de 34 —> [3, 5, 11, 17]
    card = 4
```

n = 36 –>
36 indic 12
30 indic 8
7 verifie les 3 egalites.
13 verifie les 3 egalites.
17 verifie les 3 egalites.
[7, 13, 17]
Liste des decomposants de Goldbach de 36 —> [5, 7, 13, 17]
card = 4

n = 38 –>
38 indic 18
30 indic 8
1 verifie les 3 egalites.
7 verifie les 3 egalites.
[1, 7]
Liste des decomposants de Goldbach de 38 —> [7, 19]
card = 2

n = 40 –>
40 indic 16
30 indic 8
11 verifie les 3 egalites.
17 verifie les 3 egalites.
[11, 17]
Liste des decomposants de Goldbach de 40 —> [3, 11, 17]
card = 3

n = 42 –>
42 indic 12
30 indic 8
1 verifie les 3 egalites.
11 verifie les 3 egalites.
13 verifie les 3 egalites.
19 verifie les 3 egalites.
[1, 11, 13, 19]
Liste des decomposants de Goldbach de 42 —> [5, 11, 13, 19]
card = 4

n = 44 –>
44 indic 20
30 indic 8
1 verifie les 3 egalites.
7 verifie les 3 egalites.
13 verifie les 3 egalites.
[1, 7, 13]
Liste des decomposants de Goldbach de 44 —> [3, 7, 13]
card = 3

```
n = 46 ->
    46 indic 22
    30 indic 8
    17 verifie les 3 egalites.
    [17]
    Liste des decomposants de Goldbach de 46 —> [3, 5, 17, 23]
    card = 4

n = 48 ->
    48 indic 16
    30 indic 8
    1 verifie les 3 egalites.
    7 verifie les 3 egalites.
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [1, 7, 11, 17, 19]
    Liste des decomposants de Goldbach de 48 —> [5, 7, 11, 17, 19]
    card = 5

n = 50 ->
    50 indic 20
    210 indic 48
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [13, 19]
    Liste des decomposants de Goldbach de 50 —> [3, 7, 13, 19]
    card = 4

n = 52 ->
    52 indic 24
    210 indic 48
    11 verifie les 3 egalites.
    23 verifie les 3 egalites.
    [11, 23]
    Liste des decomposants de Goldbach de 52 —> [5, 11, 23]
    card = 3
```

```
n = 54 –>
    54 indic 18
    210 indic 48
    1 verifie les 3 egalites.
    11 verifie les 3 egalites.
    13 verifie les 3 egalites.
    17 verifie les 3 egalites.
    23 verifie les 3 egalites.
    [1, 11, 13, 17, 23]
    Liste des decomposants de Goldbach de 54 —> [7, 11, 13, 17, 23]
    card = 5

n = 56 –>
    56 indic 24
    210 indic 48
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [13, 19]
    Liste des decomposants de Goldbach de 56 —> [3, 13, 19]
    card = 3

n = 58 –>
    58 indic 28
    210 indic 48
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    [11, 17]
    Liste des decomposants de Goldbach de 58 —> [5, 11, 17, 29]
    card = 4

n = 60 –>
    60 indic 16
    210 indic 48
    1 verifie les 3 egalites.
    13 verifie les 3 egalites.
    17 verifie les 3 egalites.
    19 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    [1, 13, 17, 19, 23, 29]
    Liste des decomposants de Goldbach de 60 —> [7, 13, 17, 19, 23, 29]
    card = 6
```

```
n = 62 –>
    62 indic 30
    210 indic 48
    1 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [1, 19]
    Liste des decomposants de Goldbach de 62 —> [3, 19, 31]
    card = 3

n = 64 –>
    64 indic 32
    210 indic 48
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    23 verifie les 3 egalites.
    [11, 17, 23]
    Liste des decomposants de Goldbach de 64 —> [3, 5, 11, 17, 23]
    card = 5

n = 66 –>
    66 indic 20
    210 indic 48
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    [13, 19, 23, 29]
    Liste des decomposants de Goldbach de 66 —> [5, 7, 13, 19, 23, 29]
    card = 6

n = 68 –>
    68 indic 32
    210 indic 48
    1 verifie les 3 egalites.
    31 verifie les 3 egalites.
    [1, 31]
    Liste des decomposants de Goldbach de 68 —> [7, 31]
    card = 2
```

```
n = 70 –>
    70 indic 24
    210 indic 48
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    [11, 17, 23, 29]
    Liste des decomposants de Goldbach de 70 —> [3, 11, 17, 23, 29]
    card = 5

n = 72 –>
    72 indic 24
    210 indic 48
    1 verifie les 3 egalites.
    11 verifie les 3 egalites.
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    29 verifie les 3 egalites.
    31 verifie les 3 egalites.
    [1, 11, 13, 19, 29, 31]
    Liste des decomposants de Goldbach de 72 —> [5, 11, 13, 19, 29, 31]
    card = 6

n = 74 –>
    74 indic 36
    210 indic 48
    1 verifie les 3 egalites.
    13 verifie les 3 egalites.
    31 verifie les 3 egalites.
    [1, 13, 31]
    Liste des decomposants de Goldbach de 74 —> [3, 7, 13, 31, 37]
    card = 5

n = 76 –>
    76 indic 36
    210 indic 48
    17 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    [17, 23, 29]
    Liste des decomposants de Goldbach de 76 —> [3, 5, 17, 23, 29]
    card = 5
```

n = 78 –>
78 indic 24
210 indic 48
11 verifie les 3 egalites.
17 verifie les 3 egalites.
19 verifie les 3 egalites.
31 verifie les 3 egalites.
37 verifie les 3 egalites.
[11, 17, 19, 31, 37]
Liste des decomposants de Goldbach de 78 —> [5, 7, 11, 17, 19, 31, 37]
card = 7

n = 80 –>
80 indic 32
210 indic 48
1 verifie les 3 egalites.
13 verifie les 3 egalites.
19 verifie les 3 egalites.
37 verifie les 3 egalites.
[1, 13, 19, 37]
Liste des decomposants de Goldbach de 80 —> [7, 13, 19, 37]
card = 4

n = 82 –>
82 indic 40
210 indic 48
11 verifie les 3 egalites.
23 verifie les 3 egalites.
29 verifie les 3 egalites.
[11, 23, 29]
Liste des decomposants de Goldbach de 82 —> [3, 11, 23, 29, 41]
card = 5

n = 84 –>
84 indic 24
210 indic 48
1 verifie les 3 egalites.
11 verifie les 3 egalites.
13 verifie les 3 egalites.
17 verifie les 3 egalites.
23 verifie les 3 egalites.
31 verifie les 3 egalites.
37 verifie les 3 egalites.
41 verifie les 3 egalites.
[1, 11, 13, 17, 23, 31, 37, 41]
Liste des decomposants de Goldbach de 84 —> [5, 11, 13, 17, 23, 31, 37, 41]
card = 8

```
n = 86 ->
    86 indic 42
    210 indic 48
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    [13, 19]
    Liste des decomposants de Goldbach de 86 —> [3, 7, 13, 19, 43]
    card = 5

n = 88 ->
    88 indic 40
    210 indic 48
    17 verifie les 3 egalites.
    29 verifie les 3 egalites.
    41 verifie les 3 egalites.
    [17, 29, 41]
    Liste des decomposants de Goldbach de 88 —> [5, 17, 29, 41]
    card = 4

n = 90 ->
    90 indic 24
    210 indic 48
    1 verifie les 3 egalites.
    11 verifie les 3 egalites.
    17 verifie les 3 egalites.
    19 verifie les 3 egalites.
    23 verifie les 3 egalites.
    29 verifie les 3 egalites.
    31 verifie les 3 egalites.
    37 verifie les 3 egalites.
    43 verifie les 3 egalites.
    [1, 11, 17, 19, 23, 29, 31, 37, 43]
    Liste des decomposants de Goldbach de 90 —> [7, 11, 17, 19, 23, 29, 31, 37,
43]
    card = 9

n = 92 ->
    92 indic 44
    210 indic 48
    13 verifie les 3 egalites.
    19 verifie les 3 egalites.
    31 verifie les 3 egalites.
    [13, 19, 31]
    Liste des decomposants de Goldbach de 92 —> [3, 13, 19, 31]
    card = 4
```

n = 94 –>
94 indic 46
210 indic 48
11 verifie les 3 egalites.
23 verifie les 3 egalites.
41 verifie les 3 egalites.
[11, 23, 41]
Liste des decomposants de Goldbach de 94 —> [5, 11, 23, 41, 47]
card = 5

n = 96 –>
96 indic 32
210 indic 48
13 verifie les 3 egalites.
17 verifie les 3 egalites.
23 verifie les 3 egalites.
29 verifie les 3 egalites.
37 verifie les 3 egalites.
43 verifie les 3 egalites.
[13, 17, 23, 29, 37, 43]
Liste des decomposants de Goldbach de 96 —> [7, 13, 17, 23, 29, 37, 43]
card = 7

n = 98 –>
98 indic 42
210 indic 48
1 verifie les 3 egalites.
19 verifie les 3 egalites.
31 verifie les 3 egalites.
37 verifie les 3 egalites.
[1, 19, 31, 37]
Liste des decomposants de Goldbach de 98 —> [19, 31, 37]
card = 3

n = 100 –>
100 indic 40
210 indic 48
11 verifie les 3 egalites.
17 verifie les 3 egalites.
29 verifie les 3 egalites.
41 verifie les 3 egalites.
47 verifie les 3 egalites.
[11, 17, 29, 41, 47]
Liste des decomposants de Goldbach de 100 —> [3, 11, 17, 29, 41, 47]
card = 6

Unités racines de certaines équations (Denise Vella-Chemla, 23.2.2022)

La conjecture de Goldbach stipule que tout nombre pair n supérieur ou égal à 4 est la somme $p+q$ de deux nombres premiers p et q.

On a constaté dans Retour aux unités que p et q sont à la fois des unités à n (ce sont des nombres premiers à n, i.e. des nombres x tels que $(x,n)=1$, la notation (x,n) désignant le pgcd de x et n), et des unités au produit, qu'on notera P dans la suite, de tous les nombres premiers inférieurs ou égaux à $\sqrt{n}$, c'est-à-dire qu'on a également que $\text{pgcd}(x, \prod\limits_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} p_k) = (x, \prod\limits_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} p_k) = 1$.

Gauss explique dans les *Recherches arithmétiques* que x, une unité à n (un nombre premier à n), vérifie l'équation modulaire

$$x^{\varphi(n)} \equiv 1 \pmod{n}$$

On vérifie par exemple, puisque $\varphi(98)=42$ et $\varphi(210)=48$, que 19, décomposant de Goldbach de 98, est tel que le système suivant de trois équations modulaires est vérifié :

$$\begin{cases} 19^{42} \equiv 1 \pmod{98} \\ 19^{48} \equiv 1 \pmod{210} \\ 79^{48} \equiv 1 \pmod{210} \end{cases}$$

tandis que 17 n'est pas un décomposant de Goldbach de 98 parce que $98-17=81$ et $81^{48} \not\equiv 1 \pmod{210}$.

Un décomposant de Goldbach p du nombre pair n doit donc rendre vraies les 3 équations modulaires

$$\begin{cases} p^{\varphi(n)} & \equiv 1 \pmod{n} \\ p^{\varphi(P)} & \equiv 1 \pmod{P} \\ (n-p)^{\varphi(P)} & \equiv 1 \pmod{P} \end{cases}$$

avec

$$\varphi(P) = \prod_{\substack{p_k \text{ premier} \\ 3 \leq p_k \leq \sqrt{n}}} (p_k - 1)$$

Gauss fournit un petit exemple à la toute fin de la section 92 qui clôt la section troisième *"Des résidus des puissances"* des Recherches arithmétiques :

> *"Ainsi, par exemple, pour $m = 1001 = 7.11.13$, la puissance 60 d'un nombre quelconque premier avec m, est congrue à l'unité, puisque 60 est le plus petit nombre divisible à la fois par 6, 10 et 12."*

Voici le texte complet de la section 92 des *Recherches arithmétiques* de Gauss :

> "92. Presque tout ce qui a rapport aux résidus des puissances, suivant un module composé de plusieurs nombres premiers, peut se déduire de la théorie générale des congruences ; mais comme nous exposerons plus bas une manière de ramener des congruences dont le module est composé de plusieurs nombres premiers, à d'autres dont le module est un nombre premier, ou une puissance d'un nombre premier, nous ne nous arrêterons pas beaucoup ici sur cette matière. Nous nous contenterons

d'observer que la belle propriété qui a lieu pour les autres modules, savoir : qu'il existe toujours des nombres dont la période renferme tous les nombres premiers avec le module, n'a pas lieu ici, excepté dans le seul cas où le module est le double d'un nombre premier, ou d'une puissance d'un nombre premier. En effet, si l'on ramène le module m à la forme $A^\alpha B^\beta C^\gamma$ etc., A, B, C, etc. étant des nombres premiers différents, qu'on fasse en outre $A^{a-1}(A-1) = \alpha, B^{b-1}(B-1) = \beta, C^{c-1}(C-1) = \gamma$, etc. et que z soit un nombre premier à m, on aura $z^\alpha \equiv 1 \pmod{A^a}$, $z^\beta \equiv 1 \pmod{B^b}$, etc. ; si donc μ est le plus petit nombre divisible par α, β, γ, etc., on aura $x^\mu \equiv 1$ suivant chacun des modules A^a, B^b, etc. et partant, suivant m qui est égal à leur produit ; mais excepté le cas où m est double d'un nombre premier ou d'une puissance d'un nombre premier, on a toujours $\mu < \alpha\beta\gamma$ etc., puisque les nombres α, β, etc. ne peuvent être premiers entre eux, ayant au moins le diviseur commun 2. Ainsi la période d'un nombre ne peut comprendre autant de termes qu'il y a de nombres premiers avec le module, et moindre que lui, puisque leur nombre est égal au produit $\alpha\beta\gamma$ etc. Ainsi, par exemple, pour $m = 1001 = 7.11.13$, la puissance 60 d'un nombre quelconque premier avec m, est congrue à l'unité, puisque 60 est le plus petit nombre divisible à la fois par 6, 10 et 12. Le cas où le module est double d'un nombre premier ou d'une puissance d'un nombre premier est tout à fait semblable à celui où le module est un nombre premier ou une puissance d'un nombre premier.

Considérons un nombre pair n compris entre deux carrés de nombres premiers successifs p_k^2 et p_{k+1}^2. Tous les nombres premiers susceptibles de le décomposer sont à la fois premiers à n et premiers à P le produit des nombres premiers inférieurs ou égaux à $\sqrt{n}$.

Soit p un nombre premier inférieur à $n/2$ dont on a ciblé les caractéristiques dans p-et-n-sont-toujours-de-classes-differentes.

Le complémentaire de p à n qui est égal à $n - p$ est aussi premier à $P = \prod_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} p_k$ car le plus petit commun multiple auquel Gauss fait référence, lorsque le module est le produit de nombres premiers simples, et qu'on peut noter ppcm$\{p_k - 1\}$ divise $\varphi(\prod_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} p_k) = \prod_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} (p_k - 1)$.

En effet, $\prod_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} (p_k - 1)$ contient la totalité des facteurs des factorisations des nombres $p_k - 1$ tandis que ppcm$\{p_k - 1\}$ ne contient qu'un sous-ensemble de l'ensemble des facteurs en question. Le ppcm divise l'indicateur d'Euler par simple inclusion ensembliste.

Un exemple permettra de fixer les idées : si l'on considère le produit des nombres premiers inférieurs ou égaux à 7, on a $P = 2.3.5.7$, $\varphi(P) = 1.2.4.6 = 48$ tandis que ppcm$\{1, 2, 4, 6\}$ = ppcm$\{1, 2, 2.2, 2.3\} = 12$ et $12 \mid 48$.

Ça ne va pas encore tout à fait parce que de la façon dont ça se goupille, cette condition devrait faire de tous les nombres premiers des décomposants de Goldbach d'un nombre pair donné (prenons 98, au hasard), or seuls certains le sont. On dirait que la condition est la suivante : il faut que le pgcd du ppcm$\{p_k - 1\}$ (égal à 12 pour $2.3.5.7 = 210$) et des 2 nombres $p-1$ et $q-1$ soit divisible par 6 mais on ne sait pas encore pourquoi : on a en effet que $\text{pgcd}(12, 18, 78) = 6$ (pour le décomposant de Goldbach 19 de 98) ou que $\text{pgcd}(12, 30, 66) = 6$ pour le décomposant 31 de 98, ou que $\text{pgcd}(12, 36, 60) = 6$ pour le décomposant 37 de 98. Pour 17, qui n'est pas un décomposant de 98, $\text{pgcd}(12, 16, 80) = 4$ que ne divise pas 6.

Racines de certaines équations modulaires (Denise Vella-Chemla, 1.3.2022)

La conjecture de Goldbach stipule que tout nombre pair n supérieur ou égal à 4 est la somme $p+q$ de deux nombres premiers p et q.

On rappelle que x un entier naturel est "premier à" y un autre entier naturel si leur plus grand commun diviseur est 1 (leurs factorisations ne partagent aucun facteur premier).

$$x \text{ premier à } y \iff (x,y)=1.$$

Un nombre premier est "premier à" tout autre nombre qu'il ne divise pas.

On a précisé dans *p-et-n-sont-toujours-de-classes-differentes* les caractéristiques permettant à p un nombre premier inférieur ou égal à $n/2$ d'être un décomposant de Goldbach de n (un nombre pair supérieur ou égal à 6).

Considérons n un nombre pair supérieur ou égal à 6, compris entre p_k^2 et p_{k+1}^2, p_k et p_{k+1} étant deux nombres premiers consécutifs. Notons p un nombre premier inférieur ou égal à $n/2$ et appelons $Prod$ le produit des nombres premiers inférieurs ou égaux à $\sqrt{n}$.

$$Prod = \prod_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} p_k$$

$n = p+q$ est une décomposition de Goldbach de n si on peut trouver une solution à l'équation modulaire :

$$(n-p)^{\varphi(Prod)} \equiv 1 \pmod{Prod}.$$

Gauss fournit dans la section 92 des Recherches arithmétiques la condition de résolubilité d'une telle équation (voir Annexe 1) : il faut que le plus petit commun multiple des différents nombres premiers que l'on a multipliés pour obtenir $Prod$ auxquels on soustrait 1 (i.e. ppcm$\{p_k-1\}$) divise $\varphi(Prod)$ (voir en Annexe 2 les premiers exemples numériques).

Cela découle du fait que $\varphi(\prod_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} p_k) = \prod_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} (p_k-1)$.

En effet, $\prod_{\substack{p_k \text{ premier} \\ p_k \leq \sqrt{n}}} (p_k-1)$ contient la totalité des facteurs des factorisations des nombres p_k-1 tandis que ppcm$\{p_k-1\}$ ne contient qu'un sous-ensemble de l'ensemble des facteurs en question. Le ppcm considéré divise l'indicatrice d'Euler considérée par simple inclusion ensembliste. Cette vision ensembliste de la notion de factorisation était celle de Laisant dans *Une vision ensembliste de la factorisation* .

On est donc garanti d'avoir une solution à cette équation inférieure à $Prod$.

Mais ce que l'on souhaite, c'est que cette solution soit inférieure à $n/2$. Le fait qu'il soit garanti qu'existe une solution "assez petite" de l'équation modulaire (i.e. une solution telle que p est inférieur à $n/2$) pourrait peut-être découler du théorème de Chebotarev.

Annexe 1 : Section 92 des Recherches arithmétiques de C.F. Gauss

Voici le texte complet de la section 92 des *Recherches arithmétiques* de Gauss :

> 92. Presque tout ce qui a rapport aux résidus des puissances, suivant un module composé de plusieurs nombres premiers, peut se déduire de la théorie générale des congruences ; mais comme nous exposerons plus bas une manière de ramener des congruences dont le module est composé de plusieurs nombres premiers, à d'autres dont le module est un nombre premier, ou une puissance d'un nombre premier, nous ne nous arrêterons pas beaucoup ici sur cette matière. Nous nous contenterons d'observer que la belle propriété qui a lieu pour les autres modules, savoir : qu'il existe toujours des nombres dont la période renferme tous les nombres premiers avec le module, n'a pas lieu ici, excepté dans le seul cas où le module est le double d'un nombre premier, ou d'une puissance d'un nombre premier. En effet, si l'on ramène le module m à la forme $A^\alpha B^\beta C^\gamma$ etc., A, B, C, etc. étant des nombres premiers différents, qu'on fasse en outre $A^{a-1}(A-1) = \alpha, B^{b-1}(B-1) = \beta, C^{c-1}(C-1) = \gamma$, etc. et que z soit un nombre premier à m, on aura $z^\alpha \equiv 1 \pmod{A^a}$, $z^\beta \equiv 1 \pmod{B^b}$, etc. ; si donc μ est le plus petit nombre divisible par α, β, γ, etc., on aura $x^\mu \equiv 1$ suivant chacun des modules A^a, B^b, etc. et partant, suivant m qui est égal à leur produit ; mais excepté le cas où m est double d'un nombre premier ou d'une puissance d'un nombre premier, on a toujours $\mu < \alpha\beta\gamma$ etc., puisque les nombres α, β, etc. ne peuvent être premiers entre eux, ayant au moins le diviseur commun 2. Ainsi la période d'un nombre ne peut comprendre autant de termes qu'il y a de nombres premiers avec le module, et moindre que lui, puisque leur nombre est égal au produit $\alpha\beta\gamma$ etc. Ainsi, par exemple, pour $m = 1001 = 7.11.13$, la puissance 60 d'un nombre quelconque premier avec m, est congrue à l'unité, puisque 60 est le plus petit nombre divisible à la fois par 6, 10 et 12. Le cas où le module est double d'un nombre premier ou d'une puissance d'un nombre premier est tout à fait semblable à celui où le module est un nombre premier ou une puissance d'un nombre premier.

Annexe 2 : plus petits communs multiples et indicatrices d'Euler

$Prod$	ens. des p_k1	leur ppcm	$\varphi(Prod)$	$col.4/col.3$
2.3	$\{1, 2\}$	2	2	1
2.3.5	$\{1, 2, 4\}$	4	8	2
2.3.5.7	$\{1, 2, 4, 6\}$	12	48	4
2.3.5.7.11	$\{1, 2, 4, 6, 10\}$	60	480	8
2.3.5.7.11.13	$\{1, 2, 4, 6, 10, 12\}$	60	5760	96
2.3.............17	$\{1, 2, 4, 6, 10, 12, 16\}$	240	92160	384
2.3................19	$\{1, 2, 4, 6, 10, 12, 16, 18\}$	720	1 658 880	2304
2.3..................23	$\{1, 2, 4, 6, 10, 12, 16, 18, 22\}$	7920	36 495 360	4608

Nombres p-adiquement éloignés (Denise Vella-Chemla, 14.4.2022)

Suite à la lecture du magazine Tangente Hors série n° 81, intitulé *Les distances, un outil pour tout mesurer*, on a souhaité implémenter informatiquement les distances p-adiques entre nombres entiers naturels pour étudier si cette notion nous permettrait de trouver de façon systématique les décomposants de Goldbach d'un nombre pair.

On s'est permis de transcrire en Latex les deux pages du magazine sur lesquelles on s'est appuyée pour tenter cette recherche (voir http://denise.vella.chemla.free.fr/mag-distances.pdf[1] ; noter que la distance d'un point d'une boule au centre de cette boule est maximale lorsque le rayon de la boule est maximal, c'est-à-dire lorsque le rayon de la boule est la fraction ayant pour dénominateur une puissance d'un nombre premier qui est la plus petite possible, qui est donc une puissance d'exposant le plus petit possible, nul, et la puissance est alors égale à 1 ; cela est très contre-intuitif : les entiers naturels les plus petits se retrouvent dans les boules les plus grosses, de rayons 1, voir le schéma fourni dans l'article).

Voici le programme qu'on a utilisé, pour $n = 40$. Ce programme calcule les rayons des boules auxquelles appartiennent les nombres compris entre 1 et 20, la moitié de 40, boules de centres les restes de 40 dans des divisions euclidiennes par 2, 3 et 5 (qui sont les nombres premiers inférieurs à $\sqrt{40}$).

```
n = 40
premiers = [2,3,5]
for p in premiers:
    print('')
    print('premier ', p)
    for reste in range(p):
        print(' reste = ',reste)
        puiss = 0
        while puiss <= 5:
            print(' 1/',p**puiss,'->',end='')
            moitie = int(n/2)
            for k in range(moitie):
                if k % p != 0:
                    calc = reste+k*(p**puiss)
                    if calc <= moitie:
                        print(' ',calc, end='')
            puiss = puiss+1
            print('')
```

Les distances aux centres (notées *reste* $= x$ dans le résultat du programme ci-après) pour $n = 40$ dans les différentes boules p-adiques (notées premier p dans le résultat du programme ci-après) obtenues par ce programme sont (on a supprimé les distances ne contenant pas de nombres inférieurs à $n/2$) :

[1] Je remercie Bertrand Hauchecorne pour son article. La transcription est concaténée à la fin de la présente note. Je crois qu'il y a une coquille sur l'un des deux nombres 4 dans le schéma du magazine, j'ai supprimé ce nombre, en espérant ne pas m'être trompée.

```
premier 2
    reste = 0
        1/ 1 –> 1 3 5 7 9 11 13 15 17 19
        1/ 2 –> 2 6 10 14 18
        1/ 4 –> 4 12 20
        1/ 8 –> 8
        1/ 16 –> 16
    reste = 1
        1/ 1 –> 2 4 6 8 10 12 14 16 18 20
        1/ 2 –> 3 7 11 15 19
        1/ 4 –> 5 13
        1/ 8 –> 9
        1/ 16 –> 17

premier 3
    reste = 0
        1/ 1 –> 1 2 4 5 7 8 10 11 13 14 16 17 19
        1/ 3 –> 3 6 12 15
        1/ 9 –> 9 18
    reste = 1
        1/ 1 –> 2 3 5 6 8 9 11 12 14 15 17 18 20
        1/ 3 –> 4 7 13 16
        1/ 9 –> 10 19
    reste = 2
        1/ 1 –> 3 4 6 7 9 10 12 13 15 16 18 19
        1/ 3 –> 5 8 14 17
        1/ 9 –> 11 20

premier 5
    reste = 0
        1/ 1 –> 1 2 3 4 6 7 8 9 11 12 13 14 16 17 18 19
        1/ 5 –> 5 10 15 20
    reste = 1
        1/ 1 –> 2 3 4 5 7 8 9 10 12 13 14 15 17 18 19 20
        1/ 5 –> 6 11 16
    reste = 2
        1/ 1 –> 3 4 5 6 8 9 10 11 13 14 15 16 18 19 20
        1/ 5 –> 7 12 17
    reste = 3
        1/ 1 –> 4 5 6 7 9 10 11 12 14 15 16 17 19 20
        1/ 5 –> 8 13 18
    reste = 4
        1/ 1 –> 5 6 7 8 10 11 12 13 15 16 17 18 20
        1/ 5 –> 9 14 19
```

On a étudié deux cas : celui du nombre pair 98 et celui du nombre pair 40. Dans les deux cas, on ne considère que les nombres premiers inférieurs à la racine carrée du nombre pair considéré (voir http://denise.vella.chemla.free.fr/jade1.pdf pour comprendre pourquoi cela suffit).

Prenons le cas du nombre pair 40 dont les résultats des distances p-adiques sont les résultats du programme fournis ci-dessus. On a le tableau de distances suivant. Les entêtes de colonne sont les restes modulaires de 40 ou bien le reste 0 selon chaque module premier inférieur à $\sqrt{n}$. La boule 2-adique est centrée en 0, on considère une boule 3-adique centrée en 0 et une autre centrée en 1, la boule 5-adique est centrée en 0. Une croix dans la colonne DG indique que le nombre entête de ligne est un décomposant de Goldbach de n.

	0 (mod 2)	0 (mod 3)	1 (mod 3)	0 (mod 5)	*DG*
3	1/1	1/3	1/1	1/1	
5	1/1	1/1	1/1	1/5	
7	1/1	1/1	1/3	1/1	
9	1/1	1/9	1/1	1/1	
11	1/1	1/1	1/1	1/1	$*$
13	1/1	1/1	1/3	1/1	
15	1/1	1/3	1/1	1/5	
17	1/1	1/1	1/1	1/1	$*$
19	1/1	1/1	1/9	1/1	

Les 3 décomposants de Goldbach que sont les nombres 3, 11 et 17 maximisent les distances aux centres dans chacune des boules, centrées sur les restes de 40 ou sur 0.

Pour 98, on obtient les distances suivantes aux restes modulaires de 98 ou aux centres nuls, indiqués en tête des colonnes : une case de la colonne d'entête r_k (mod p) fournit la distance p-adique du nombre entête de ligne au centre de la boule (et donc au rayon de la boule) à laquelle ce nombre appartient. La boule en question est centrée en r_k pour k compris entre 0 et $p-1$, r_k étant le reste modulaire considéré.

	0 (mod 2)	0 (mod 3)	2 (mod 3)	0 (mod 5)	3 (mod 5)	0 (mod 7)	*DG*
3	1/1	1/3	1/1	1/1	0 ?	1/1	
5	1/1	1/1	1/3	1/5	1/1	1/1	
7	1/1	1/1	1/1	1/1	1/1	1/7	
9	1/1	1/9	1/1	1/1	1/1	1/1	
11	1/1	1/1	1/9	1/1	1/1	1/1	
13	1/1	1/1	1/1	1/1	1/5	1/1	
15	1/1	1/3	1/1	1/5	1/1	1/1	
17	1/1	1/1	1/3	1/1	1/1	1/1	
19	1/1	1/1	1/1	1/1	1/1	1/1	$*$
21	1/1	1/3	1/1	1/1	1/1	1/7	
23	1/1	1/1	1/3	1/1	1/5	1/1	
25	1/1	1/1	1/1	1/25	1/1	1/1	
27	1/1	1/27	1/1	1/1	1/1	1/1	
29	1/1	1/1	1/27	1/1	1/1	1/1	
31	1/1	1/1	1/1	1/1	1/1	1/1	$*$
33	1/1	1/3	1/1	1/1	1/5	1/1	
35	1/1	1/1	1/3	1/5	1/1	1/7	
37	1/1	1/1	1/1	1/1	1/1	1/1	$*$
39	1/1	1/3	1/1	1/1	1/1	1/1	
41	1/1	1/1	1/3	1/1	1/1	1/1	
43	1/1	1/1	1/1	1/1	1/5	1/1	
45	1/1	1/9	1/1	1/5	1/1	1/1	
47	1/1	1/1	1/9	1/1	1/1	1/1	
49	1/1	1/1	1/1	1/1	1/1	1/49	

On voit que les 3 décomposants de Goldbach de 98 que sont 19, 31 et 37 maximisent la distance dans les 6 boules centrées sur les restes modulaires de 98 ou bien sur 0[2]. On voit également que les

[2]On s'était intéressée, en 2015 http://denise.vella.chemla.free.fr/distancesupreme1.pdf et en mai 2016

distances “amenuisées” dans les colonnes des restes de n indiquent les diviseurs du complémentaire du nombre premier considéré (entête d’une ligne) à 98.

Ces “intersections de boules topologiques” permettraient peut-être d’établir l’existence obligatoire d’un décomposant de Goldbach pour tout nombre pair supérieur ou égal à 6. Un problème est que les rayons sont incommensurables d’une colonne à l’autre. Peut-être faudrait-il utiliser une propriété surprenante qui est que dans un espace ultramétrique, tout triangle est isocèle, en considérant systématiquement les nombres 3 par 3.

---

http://denise.vella.chemla.free.fr/tore.pdf à l’idée de distance suprême entre nombres premiers, dans des corps de restes, sans succès. Cette idée de considérer une distance suprême provenait de la lecture, sans maîtrise, du livre *Noncommutative geometry* d’Alain Connes (https://alainconnes.org/wp-content/uploads/book94bigpdf.pdf), qui définit la distance comme un supremum à la page 35. Il est également question d’une distance ultramétrique à la page 10 de cette traduction qu’on a faite (http://denise.vella.chemla.free.fr/trad-bc-source.pdf) de l’article d’Alain Connes et Jean-Benoît Bost *Hecke Algebras, Type III Factors and Phase Transitions with Spontaneous Symmetry Breaking in Number Theory*, Selecta Math. (N.S.), vol. 1, no. 3, pp. 411–457, 1995, ISSN: 1022-1824, https://alainconnes.org/wp-content/uploads/bostconnesscan.pdf, mais on n’a pas le bagage mathématique suffisant pour voir si cela correspondrait aux mêmes distances que celles qui sont calculées ici.

Pages 32 et 33 du magazine Tangente (Hors série n° 81) sur les distances (Denise Vella-Chemla, 9.4.2022)

Les distances ultramétriques

Considérons un ensemble E et définissons, pour tous éléments a et b de E, $\delta(a,b) = 1$ si a est différent de b et $\delta(a,a) = 0$. En quelque sorte, il s'agit d'un indicateur booléen qui teste si deux éléments sont distincts on non. Il vérifie clairement les axiomes de positivité, de symétrie et de coïncidence.

Prenons trois éléments a, b et c de E. Si a est différent de c, alors $\delta(a,c) = 1$. Or soit a est différent de b et $\delta(a,b) = 1$, soit b est différent de c et $\delta(b,c) = 1$, soit les deux. Alors $\delta(a,c) = 1 \leq \max(\delta(a,b), \delta(b,c)) \leq \delta(a,b) + \delta(b,c)$. Ainsi, l'axiome du triangle est vérifié. Mais on a obtenu en fait un résultat bien plus fort, plus restrictif. De manière plus générale, on parle de distance ultramétrique lorsque l'axiome du triangle est remplacé par le suivant, plus exigeant, affirmant que pour tous les éléments a, b et c de E on a $\delta(a,c) \leq \max(\delta(a,b), \delta(b,c))$.

De tels espaces *ultramétriques* recèlent des propriétés qui défient l'intuition. Dans notre exemple, la boule ouverte et de rayon 1, centrée en un point a quelconque, ne contient que le point a, tandis que la boule fermée associée englobe tout l'espace, donc contient toutes les boules ouvertes de rayon 1, quel que soit leur centre.

De manière plus générale, dans un espace ultramétrique, les boules ouvertes sont des parties fermées et tout point d'une boule en est le centre (voir encadré). Autre particularité des espaces ultramétriques, tous les triangles y sont isocèles ! Pour le montrer, considérons trois éléments a, b et c de l'espace ultramétrique E.

Supposons que le triangle de sommets a, b et c ne soit pas équilatéral et que $\delta(a,b) < \delta(b,c)$. Alors $\delta(a,c) = \delta(b,c)$ car sinon le plus grand de ces deux termes ne vérifierait pas l'inégalité ultramétrique, étant strictement plus grand que chacun des deux autres. Ainsi, si le triangle n'est pas équilatéral, le côté inégal aux deux autres est nécessairement le plus petit.

Les espaces ultramétriques se sont révélés particulièrement intéressants (voir Tangente 190, 2019), en particulier pour l'étude des distances p-adiques dans $\mathbb{Z}$ et $\mathbb{Q}$ (voir encadré). En voici un exemple. Considérons un ensemble G et les suites d'éléments de G indexées sur les entiers strictement positifs. Considérons deux suites distinctes $A = a_1, a_2, a_3, \ldots$ et $B = b_1, b_2, b_3, \ldots$ Posons $\delta(A,B) = 1$ si $a_1 \neq b_1$, $\delta(A,B) = 1/2$ si $a_1 = b_1$ et $a_2 \neq b_2$, et plus généralement $\delta(A,B) = 1/n$ si n désigne le plus petit indice tel que $a_1 = b_1, a_2 = b_2, \ldots, a_{n-1} = b_{n-1}$ et $a_n \neq b_n$. En ajoutant la convention $\delta(A,A) = 0$ pour toute suite A, on obtient une distance ultramétrique. Vérifions le troisième axiome en prenant trois suites A, B et C avec $C = c_1, c_2, c_3, \ldots$. Si $\delta(A,C) = 1/n$, alors $a_n \neq c_n$ donc soit $a_n \neq b_n$, soit $b_n \neq c_n$. Cela revient à dire que soit $\delta(A,B) \geq 1/n$, soit $\delta(B,C) \geq 1/n$. Finalement, $\delta(A,C) = 1/n \leq \max(\delta(A,B), \delta(B,C))$. CQFD.

En prenant des ensembles G différents, on obtient de nombreux exemples ; en voici un dans le domaine de la linguistique. Prenons pour G l'ensemble $\{a, b, c,$ ç$, d, e,$ é, è, ê$, ..., z, -, \omega\}$, c'est-à-dire

toutes les lettres (y compris accentuées...) utilisées en français, auxquelles on ajoute un symbole ω pour représenter les espaces. Tout mot de notre langue, et même toute phrase, est une suite d'éléments de G, à condition de terminer par des ω pour la rendre infinie. On définit ainsi avec δ, comme vu précédemment, une distance sur les mots de notre langue.
Prenons A = "barrette", B = "barreau" et C = "barrique" ; on obtient $\delta(A, B) = 1/6$ puisque les mots "barrette" et "barreau" diffèrent seulement à la sixième lettre tandis que $\delta(A, C) = \delta(B, C) = 1/5$ puisque le mot "barrique" diffère dès la cinquième lettre des deux autres mots. Comme on le voit, ces trois éléments forment un triangle équilatéral pour la distance δ.

Le concept d'espace métrique, élaboré dans l'esprit fertile de Maurice Fréchet, étayé par le sérieux germanique de Felix Hausdorff, avait pour objectif de généraliser la notion de limite dans des espaces de fonctions. Comme souvent en mathématiques, cette structure s'est avérée féconde dans des domaines les plus variés.

Topologie d'un espace métrique

Dans un espace métrique E, lorsqu'une boule ouverte contient a, les points immédiatement "proches" de a y appartiennent également. Cette propriété est intéressante pour étudier une fonction au voisinage d'un point. C'est pourquoi on dit qu'une partie A de E est *ouverte* si, pour tout point a de A, il existe une boule centrée en a contenue entièrement dans A : clairement, les boules ouvertes sont ouvertes (ouf !). Ce n'est pas une tautologie, mais une adéquation des définitions.

Si une suite d'éléments d'une boule fermée $B'(a, r)$ converge, sa limite appartient encore à cette boule. Plus généralement, on appelle *partie fermée* celle vérifiant cette propriété (et les boules fermées sont donc fermées, ouf !).

Des boules ouvertes et fermées

En général, dans un espace métrique, les boules ouvertes ne sont pas fermées. C'est pourtant le cas dans les espaces ultramétriques. Soit en effet une suite convergente $(x_n)_{n\leq 0}$ appartenant à la boule ouverte $B(a, r)$ et soit L sa limite. Alors $\delta(a, L) \leq \max(\delta(a, x), \delta(x, L))$. En prenant n "assez grand" pour que $\delta(x_n, L) < r$, on obtient $\delta(a, L) < r$, ce qui montre que L appartient à $B(a, r)$: cette boule ouverte est fermée ! Fermez le ban.

De même, tout point d'une boule est le centre de cette boule. En effet, considérons la boule $B(a, r)$ de centre a et de rayon r et un point b dans cette boule. Alors, pour tout x de $B(a, r)$, on a $\delta(b, x) \leq \max(\delta(b, a), \delta(a, x)) < r$ puisque b comme r appartiennent à la boule $B(a, r)$. En inversant les rôles de a et b, on a l'égalité $B(a, r) = B(b, r)$: tous les points de la boule sont donc des centres. On en conclut que deux boules ne peuvent pas être "sécantes" : si elles ne sont pas disjointes, l'une des deux contient nécessairement l'autre.

Distances p-adiques sur les entiers

Prenons un nombre premier p. Tout entier relatif non nul s'écrit de manière unique sous la forme $a = \pm b \times p^k$, où k est la plus grande puissance de p qui divise a et b désigne un nombre premier avec p. Ainsi, en prenant $p = 3$, on a par exemple $2 = 2 \times 3^0, 14 = 14 \times 3^0$ ou encore $315 = 35 \times 3^2$. On note $v_p(a) = k$ pour signifier que la plus grande puissance de p qui divise a est k. Si a et a' sont deux entiers, alors $v_p(a + a') \geq \min(v_p(a), v_p(a'))$ et $v_p(a \times a') = v_p(a) + v_p(a')$.

Posons alors $\delta(a, a') = p^{-v_p(a-a')}$ si $a \neq a'$ et $\delta(a, a) = 0$. Cette quantité est clairement positive. Elle n'est nulle que si $a = a'$. Manifestement, $\delta(a, a') = \delta(a', a)$ puisque $v_p(a - a') = v_p(a' - a)$. Enfin, l'inégalité ultramétrique découle de l'inégalité vue plus haut. Reprenons $p = 3$ et considérons une boule ouverte centrée en 2. La distance ne prenant, par définition de δ, que des valeurs de la forme $r_n = 1/3^n$, les boules ouvertes sont fermées, et réciproquement.
Supposons que $3^{-(n+1)} \leq r < 3^{-n}$; alors la boule $B(a, r)$ contient les éléments dont la différence avec 2 est divisible par 3^n, c'est-à dire les termes de la forme $2 + 3^n \times u$ où u désigne un entier relatif.

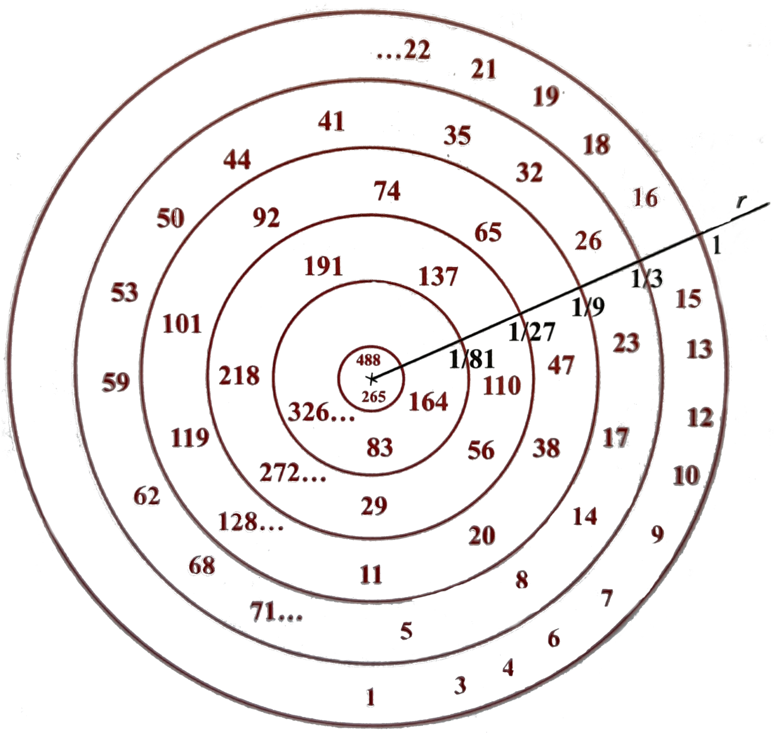


Figure 1 : Boules fermées de centre 2 et de rayons $r = 1/81, 1/27, 1/9, 1/3$ et 1 pour la distance p-adique.

δ s'appelle la *distance p-adique* sur l'ensemble $\mathbb{Z}$ des entiers relatifs. Elle se généralise sans difficulté à tous les rationnels, en posant, pour tout nombre rationnel de la forme $r = a/b$ avec a et b entiers, b non nul, $v_p(r) = v_p(a) - v_p(b)$.

La convergence des suites peut surprendre. On sait que si un nombre réel t est compris strictement entre -1 et 1, alors la suite définie par $t_n = 1 + t + t^2 + ... + t^n$ converge vers $1/(1 - t)$. Dans le cadre des nombres p-adiques, cette formule peut subsister pour des nombres plus grands que 1 ! Prenons toujours $p = 3$ et la suite $T_n = 1 + 3 + 3^2 + ... + 3^n$; alors $1/(1 - 3) = -1/2$.

Montrons que ce nombre est bien la limite de notre suite. Comme $T_n = \dfrac{1 - 3^{n+1}}{1 - 3} = -\dfrac{1}{2} + \dfrac{3^{n+1}}{2}$ pour tout entier n, on a aussi $T_n - \left(-\frac{1}{2}\right) = \dfrac{3^{n+1}}{2}$. Or, $\nu_3\left(\dfrac{3^{n+1}}{2}\right) = \nu_3(3^{n+1}) - \nu_3(2) = n + 1 - 0 = n + 1$

donc $\delta\left(T_n, -\frac{1}{2}\right) = 3^{-n-1}$, quantité qui tend vers 0 quand n tend vers l'infini. Ceci montre que la suite $(T_n)_{n\geq 0}$ tend vers -1/2.

Petits triangles (Denise Vella-Chemla, 11.5.2022)

Dans une note précédente[1], on a démontré qu'un nombre premier supérieur à $\sqrt{n}$ et inférieur ou égal à $n/2$, et qui n'est jamais congru à n modulo tout nombre premier inférieur à $\sqrt{n}$ est un décomposant de Goldbach de n.

Il s'agit de démontrer qu'un tel nombre existe toujours.

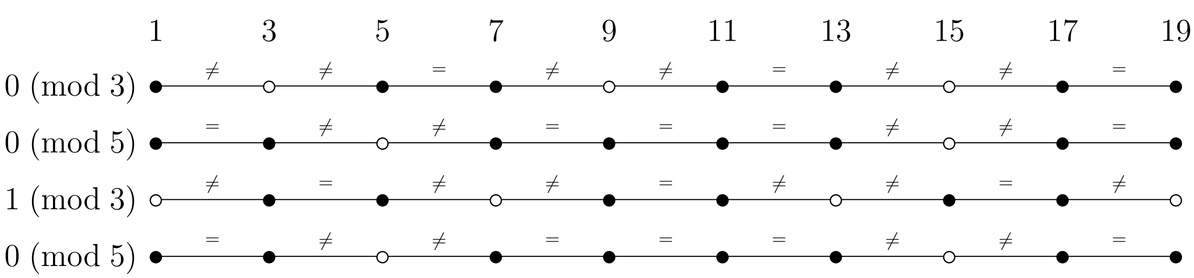
À l'aide d'un exemple (la recherche des décomposants de Goldbach du nombre pair 40), on montre l'idée topologique qu'on suit en ce moment.

On aimerait utiliser la distance triviale (ou distance discrète, voir définition en annexe) et le fait que tout triangle est isocèle dans un espace ultramétrique.

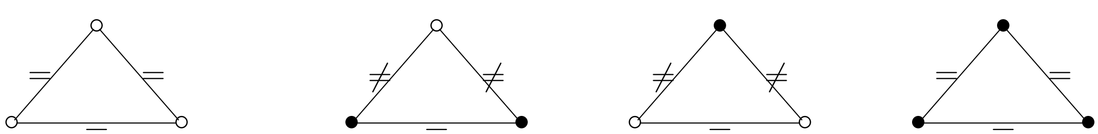
À la recherche des décomposants de Goldbach de 40, on commence par dessiner des lignes de nœuds, reliés par des arcs.

- les 2 premières lignes de nœuds correspondent aux relations entre les nombres impairs (notés sur une ligne à part en haut, en regard des lignes de nœuds, et le reste nul selon tout module premier impair inférieur à la racine de n (ici, 3 et 5, inférieurs à $\sqrt{40} \simeq 6, \ldots$) ; ces lignes font apparaître la divisibilité par tel ou tel nombre premier ;
- les 2 lignes suivantes montrent le fait qu'un nombre a le même reste que le nombre $n = 40$ selon ces mêmes nombres premiers choisis comme modules.

Remarque : comme 5 divise 40, il y a redondance entre la deuxième et la quatrième ligne, on maintient cette redondance, pour avoir exactement $2\,(\pi(\sqrt{n}) - 1)$ lignes, simplement ($\pi(x)$ représente comme habituellement le nombre de nombres premiers inférieurs ou égaux à x et on soustrait 1 parce qu'on oublie 2 en se concentrant sur les impairs, pour des raisons de lisibilité de l'exemple).



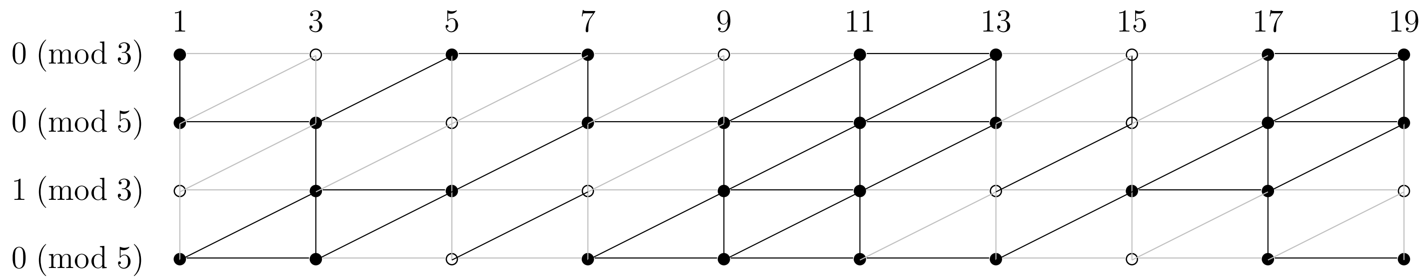
Dans une seconde étape, triangulons : dans un espace ultramétrique muni de la distance triviale, tout triangle est isocèle. Les 4 triangles possibles sont :



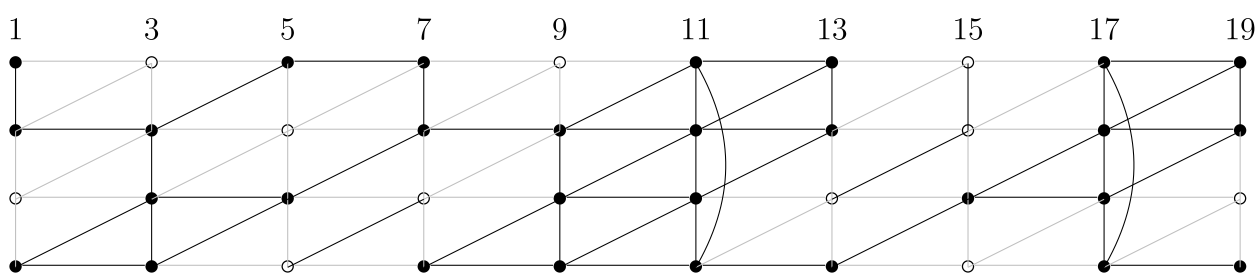
[1] http://denise.vella.chemla.free.fr/jade1.pdf.

Ces petits triangles correspondent aux phrases : “deux nombres égaux à un même troisième sont égaux”, “si deux nombres sont différents, un nombre égal à l’un de ces deux nombres est différent de l’autre de ces deux nombres”.

Voici le résultat de l’action de triangulation appliquée aux 3 lignes initiales. Pour améliorer la lisibilité du graphe, on a remplacé les arcs étiquetés par le signe $=$ par des arcs noirs et ceux étiquetés par le signe $\neq$ par des arcs gris clair.



Pour finir, on ajoute des “arcs-ponts” verticalement “partout” (ce qui ne sera pas fait ci-après), parce que cela permettra de ramener la conjecture de Goldbach à la recherche d’un cycle. Montrons les seuls ajouts de 2 arcs-ponts, d’une part pour ne pas surcharger le dessin, d’autre part pour voir les deux décomposants de Goldbach de 40 supérieurs à $\sqrt{40}$ que sont 11 et 17.



Démontrer la conjecture de Goldbach consiste alors à démontrer l’existence d’un cycle de longueur $2(\pi(\sqrt{n}) - 1)$ dans le graphe, cycle dont tous les arcs seraient étiquetés par le signe $=$. Mais cette condition ne suffit pas, on voit dans cet exemple un certain nombre de parallélogrammes de côtés colorés en noir qui sont autant de cycles. Il faudrait caractériser le fait que les nœuds du cycle sont dans une configuration verticale du graphe (i.e. correspondent à un même nombre). De plus, on ne sait pas démontrer l’existence d’un tel cycle.

Annexe : définitions

On recopie ici les définitions de la distance triviale et de la topologie discrète trouvées sur wikipedia.

La distance triviale (ou encore distance discrète ou métrique discrète) est définie sur tout ensemble par : $d(x, y) = 1$ si $x \neq y$ et $d(x, x) = 0$. La topologie associée est la topologie discrète.

En topologie, la topologie discrète sur un ensemble est une structure d'espace topologique où, de façon intuitive, tous les points sont "isolés" les uns des autres.

Soit X un ensemble. L'ensemble des parties de X définit une topologie sur X appelée topologie discrète. X muni de cette topologie est alors appelé espace discret.

On dit qu'une partie A d'un espace topologique X est un ensemble discret lorsque la topologie induite sur A est la topologie discrète.

Expression de la conjecture de Goldbach en logique du premier ordre (Denise Vella-Chemla, 15.5.2022)

Dans une note précédente[1], on a démontré qu'un nombre premier supérieur à $\sqrt{n}$ et inférieur ou égal à $n/2$, et qui n'est jamais congru à n modulo tout nombre premier inférieur à $\sqrt{n}$ est un décomposant de Goldbach de n.

Il s'agit de démontrer qu'un tel nombre existe toujours.

Si on suivait cette façon de voir la conjecture, on pourrait peut-être la démontrer en réussissant à faire démontrer à un ordinateur (en utilisant Lean, Coq ou Isabelle) l'énoncé de la logique du premier ordre ci-dessous :

$$\begin{array}{l} \forall\, n \in \mathbb{N}, n \geqslant 6 \wedge 2 \mid n, \\ \exists\, p \in \mathbb{N}, p \leqslant \frac{n}{2}, \\ \qquad \forall\, q \in \mathbb{N}, q \leqslant \sqrt{n} \ \text{ tq } \ \forall\, r \leqslant \sqrt{q},\ \ r \nmid q, \\ \qquad\qquad p \not\equiv 0 \pmod{q} \wedge\ p \not\equiv n \pmod{q}. \end{array}$$

Si on veut se passer de la notation par les congruences de Gauss ($x \equiv y \pmod{m}$), il faut aussi dire à l'ordinateur que :

$$\begin{array}{l} \forall\, a \in \mathbb{N}, \forall\, r' \in \mathbb{N}, \forall\, q \in \mathbb{N}^*, \\ \qquad a \not\equiv r' \pmod{q} \iff \exists\, r \in [0, q-1] \ \text{ tq } \ a = bq + r \wedge r \neq r'. \end{array}$$

[1] http://denise.vella.chemla.free.fr/jade1.pdf.

Calcule et ... (Denise Vella-Chemla, juillet 2022)

On reprend notre exemple fétiche de la recherche des décomposants de Goldbach de l'entier pair $n = 98$.

$$S_{98} = \begin{cases} 98 \equiv 0 \pmod 2 \\ 98 \equiv 2 \pmod 3 \\ 98 \equiv 3 \pmod 5 \\ 98 \equiv 0 \pmod 7 \end{cases}$$

Appelons d_{98} un décomposant de Goldbach potentiel de $n = 98$. d_{98} peut être congru, hormis 0, à tout ce à quoi $n = 98$ n'est pas congru. Le signe $\vee$ dans le système ci-dessous est à lire comme un ou exclusif, son emploi étendu est à comprendre comme le fait de vérifier autant de systèmes de congruences que la combinatoire le permet.

$$S_{d_{98}} = \begin{cases} d_{98} \equiv 1 \pmod 2 \\ d_{98} \equiv 1 \pmod 3 \\ d_{98} \equiv 1 \vee 2 \vee 4 \pmod 5 \\ d_{98} \equiv 1 \vee 2 \vee 3 \vee 4 \vee 5 \vee 6 \pmod 7 \end{cases}$$

Remarque 1 : on note que la congruence à 1 mod 2 garantit que la solution est un nombre impair.

Remarque 2 : on notera que le fait de respecter le système de systèmes de congruences ci-dessus est une condition suffisante mais non nécessaire pour être un décomposant de Goldbach de n. On trouvera la preuve de cette caractérisation des décomposants de Goldbach d'un nombre pair n qui sont supérieurs à la racine carrée de n en [1].

Comme on peut le comprendre, les modules qui ne divisent pas n "éliminent davantage de classes de congruences" (au nombre de 2) que les modules qui divisent n. Plaçons-nous dans le pire des cas, où l'on élimine deux classes de congruences par module premier inférieur à $\sqrt{n}$, on trouve tout de même

$$\frac{1}{2} \prod_{\substack{p \text{ premier} \\ 5 \leq p \leq \sqrt{n}}} (p-2)$$

classes de congruences différentes par l'application du théorème des restes chinois à chacun des systèmes de congruences combinatoirement trouvé (voir $S_{d_{98}}$ ci-dessus). La division par 2 est justifiée par les symétries autour des moitiés (par exemple, pour 40, les classes de congruences trouvées par l'application du théorème des restes chinois à chaque système[1] sont les classes $30k+11, 30k+17, 30k+23$ et $30k+29$ dont on ne conserve que la moitié par symétrie autour de 20, la moitié de 40.

Mais d'autre part, les solutions étant toutes des unités du groupe $(\mathbb{Z}/n\mathbb{Z})^\times$, la moitié d'entre elles sont inférieures à $D = \frac{1}{2} \prod_{\substack{p \text{ premier} \\ 3 \leq p \leq \sqrt{n}}} p$ (pour illustrer cela sur l'exemple $n = 98$, la moitié des solutions

[1]Systèmes de congruences pour $n = 40$:

$$\begin{cases} 40 \equiv 0 \pmod 2 \\ 40 \equiv 1 \pmod 3 \\ 40 \equiv 0 \pmod 5 \end{cases} \qquad S_{d_{40}} = \begin{cases} d_{40} \equiv 1 \pmod 2 \\ d_{40} \equiv 2 \pmod 3 \\ d_{40} \equiv 1 \vee 2 \vee 3 \vee 4 \pmod 5 \end{cases}$$

(s'il en existe) sont forcément inférieures à $105=3\times 5\times 7$).

Serait-il possible de "rater l'intervalle visé", i.e. que toutes les solutions soient supérieures à n, comprises entre n et D ?

Oui, ce serait tout à fait possible : le nombre de solutions trouvées combinatoirement étant très vite inférieur au nombre d'impairs compris entre n et D qui est égal à $\dfrac{D-n+1}{2}$, cette approche est nulle et non avenue.

Référence

[1] D. Chemla, *Réécrire*, http://denise.vella.chemla.free.fr/jade1.pdf

Équirépartition dans le groupe des unités ? (Denise Vella-Chemla 19.12.2021)

Revenons une fois encore à notre exemple fétiche de la recherche des décomposants de Goldbach de 98 pour voir ce qui se passe “derrière la scène” (cf. [2], [3], [4]).

On a besoin de considérer les seuls nombres premiers inférieurs à la racine carrée de 98, soient 2, 3, 5 et 7.

98 vérifie les congruences suivantes : $98 \equiv 0\ [2], 98 \equiv 2\ [3], 98 \equiv 3\ [5], 98 \equiv 0\ [7]$.

Un décomposant de Goldbach de 98 supérieur à $\sqrt{98}$, s'il existe, devra vérifier les congruences $x \equiv 1\ [2], x \equiv 1\ [3], x \equiv 1 \vee 2 \vee 4\ [5], x \equiv 1 \vee 2 \vee 3 \vee 4 \vee 5 \vee 6\ [7]$. Ci-dessus, les modules sont indiqués entre crochets et la lettre $\vee$ symbolise une disjonction de congruences.

On voit qu'un décomposant de Goldbach de 98 supérieur à $\sqrt{98}$ est la solution potentielle d'un système de congruences parmi $1 \times 1 \times 3 \times 6 = 18$ systèmes de congruences possibles.

Chacun de ces systèmes de congruences se résout en utilisant le théorème des restes chinois.

Chaque solution parmi les 18 possibles est un nombre de la forme suivante (ce nombre est calculé modulo 210, le produit de tous les nombres premiers inférieurs à $\sqrt{98}$, il représente une infinité de nombres possibles) :

$$\begin{array}{rl}
& \Bigg(\underbrace{[1\ .\ (3.5.7)]}_{\substack{\equiv 1\ [2] \\ \equiv 0\ [3] \\ \equiv 0\ [5] \\ \equiv 0\ [7]}} \times a_1 + \underbrace{[1\ .\ (2.5.7)]}_{\substack{\equiv 0\ [2] \\ \equiv 1\ [3] \\ \equiv 0\ [5] \\ \equiv 0\ [7]}} \times a_2 + \underbrace{[3\ .\ (2.3.7)]}_{\substack{\equiv 0\ [2] \\ \equiv 0\ [3] \\ \equiv 1\ [5] \\ \equiv 0\ [7]}} \times a_3 + \underbrace{[4\ .\ (2.3.5)]}_{\substack{\equiv 0\ [2] \\ \equiv 0\ [3] \\ \equiv 0\ [5] \\ \equiv 1\ [7]}} \times a_4 \Bigg) \ \ [2.3.5.7] \\
= & \Big(105\ a_1 + 70\ a_2 + 126\ a_3 + 120\ a_4 \Big) \ \ [210]
\end{array}$$

Comme a_1 (resp. a_2) ne peut prendre que la valeur 1, 105 a_1 (resp. 70 a_2) est égal à 105 (resp. 70).

Comme a_3 ne peut prendre que les valeurs 1 ou 2 ou 4, 126 a_3 est égal à $126, 252$ ou 504.

Comme a_4 peut prendre n'importe quelle valeur de 1 à 6, 120 a_4 est égal à 120 ou 240 ou 360 ou 480 ou 600 ou 720.

En combinant toutes ces possibilités (au nombre de 18), voici les valeurs que prend $105\ a_1 + 70\ a_2 + 126\ a_3 + 120\ a_4$ avant réduction modulo 210 (on note les quadruplets (a_1, a_2, a_3, a_4) en regard de chaque valeur). Les décomposants de Goldbach de 98 et leur complémentaire sont notés en bleu.

$$\begin{array}{lll}
(1,1,1,1) \to 421 & (1,1,2,1) \to 547 & (1,1,3,1) \to 799 \\
(1,1,1,2) \to 541 & (1,1,2,2) \to 667 & (1,1,3,2) \to 919 \\
(1,1,1,3) \to 661 & (1,1,2,3) \to 787 & (1,1,3,3) \to 1039 \\
(1,1,1,4) \to 781 & (1,1,2,4) \to 907 & (1,1,3,4) \to 1159 \\
(1,1,1,5) \to 901 & (1,1,2,5) \to 1027 & (1,1,3,5) \to 1279 \\
(1,1,1,6) \to 1021 & (1,1,2,6) \to 1147 & (1,1,3,6) \to 1399
\end{array}$$

Voici les mêmes nombres réduits modulo 210.

1	127	169
121	37	79
31	157	199
151	67	109
61	187	19
181	97	139

Comme attendu, ces nombres sont tous des unités du groupe $\mathbb{Z}/210\mathbb{Z}$ car leur expression montrent qu'ils ne sont divisibles ni par 2, ni par 3, ni par 5, ni par 7.

$\varphi(210) = 48$ ($\varphi(n)$ compte les unités de $\mathbb{Z}/210\mathbb{Z}$, le nombre de nombres inférieurs à n et premiers à n).

Les unités trouvées comme solutions potentielles par le théorème des restes chinois, au nombre de 18 sur 48 unités, sont équiréparties dans l'intervalle $[3, 210]$.

Cette équirépartition a pour conséquence que 3 systèmes de congruences parmi les 18 systèmes envisageables ont pour solution respective les nombres $19, 31$ et 37, qui sont tous les trois compris entre 3 et 49 la moitié de 98 et sont donc des décomposants de Goldbach de 98.

Si l'on cherche maintenant les décomposants de Goldbach de 120 ($\equiv 0\,[2], \equiv 0\,[3], \equiv 0\,[5], \equiv 1\,[7]$), lui aussi compris entre 7^2 et 11^2, une solution potentielle a davantage de latitude ($\equiv 1\,[2], \equiv 1 \vee 2\,[3], \equiv 1 \vee 2 \vee 3 \vee 4\,[5], \equiv 2 \vee 3 \vee 4 \vee 5 \vee 6\,[7]$). Les solutions potentielles à calculer modulo 210 sont au nombre de 40 et l'intervalle à couvrir est plus grand $[3, 60]$. Le nombre ramené dans $\mathbb{Z}/210\mathbb{Z}$ est noté à côté de chaque $105a_1 + 70a_2 + 126a_3 + 120a_4$, après une virgule. On a coloré en bleu les décomposants de Goldbach et leur complémentaire pour 120 et 104 dans les tableaux ci-après.

$(1,1,1,2) \to 541, 121$	$(1,2,1,2) \to 611, 191$
$(1,1,1,3) \to 661, 31$	$(1,2,1,3) \to 731, 101$
$(1,1,1,4) \to 781, 151$	$(1,2,1,4) \to 851, 11$
$(1,1,1,5) \to 901, 61$	$(1,2,1,5) \to 971, 131$
$(1,1,1,6) \to 1021, 181$	$(1,2,1,6) \to 1091, 41$
$(1,1,2,2) \to 667, 37$	$(1,2,2,2) \to 737, 107$
$(1,1,2,3) \to 787, 157$	$(1,2,2,3) \to 857, 17$
$(1,1,2,4) \to 907, 67$	$(1,2,2,4) \to 977, 137$
$(1,1,2,5) \to 1027, 187$	$(1,2,2,5) \to 1097, 47$
$(1,1,2,6) \to 1147, 97$	$(1,2,2,6) \to 1217, 167$
$(1,1,3,2) \to 793, 163$	$(1,2,3,2) \to 863, 23$
$(1,1,3,3) \to 913, 73$	$(1,2,3,3) \to 983, 143$
$(1,1,3,4) \to 1033, 193$	$(1,2,3,4) \to 1103, 53$
$(1,1,3,5) \to 1153, 103$	$(1,2,3,5) \to 1223, 173$
$(1,1,3,6) \to 1273, 13$	$(1,2,3,6) \to 1343, 83$
$(1,1,4,2) \to 919, 79$	$(1,2,4,2) \to 989, 149$
$(1,1,4,3) \to 1039, 199$	$(1,2,4,3) \to 1109, 59$
$(1,1,4,4) \to 1159, 109$	$(1,2,4,4) \to 1229, 179$
$(1,1,4,5) \to 1279, 19$	$(1,2,4,5) \to 1349, 89$
$(1,1,4,6) \to 1399, 139$	$(1,2,4,6) \to 1469, 209$

On peut constater en étudiant le cas $n = 104 = 8 \times 13$ que les solutions, bien qu'étant des nombres premiers qui donc sont premiers à n, ne sont cependant pas à chercher dans le groupe des unités de $\mathbb{Z}/n\mathbb{Z} = \mathbb{Z}/104\mathbb{Z}$ mais dans le groupe des unités de $\mathbb{Z}/(2.3.5.7)\mathbb{Z} = \mathbb{Z}/210\mathbb{Z}$.

On a $104 \equiv 0\ [2], \equiv 2\ [3], \equiv 4\ [5], \equiv 6\ [7]$.

Une solution potentielle vérifie $x \equiv 1\ [2], \equiv 1\ [3], \equiv 1 \vee 2 \vee 3\ [5], \equiv 1 \vee 2 \vee 3 \vee 4 \vee 5\ [7]$, il y a 15 solutions potentielles à calculer modulo 210, l'intervalle est $[3, 52]$.

$(1,1,1,1) \to 421, 1$	$(1,1,2,1) \to 547, 127$	$(1,1,3,1) \to 673, 43$
$(1,1,1,2) \to 541, 121$	$(1,1,2,2) \to 667, 37$	$(1,1,3,2) \to 793, 163$
$(1,1,1,3) \to 661, 31$	$(1,1,2,3) \to 787, 157$	$(1,1,3,3) \to 913, 73$
$(1,1,1,4) \to 781, 151$	$(1,1,2,4) \to 907, 67$	$(1,1,3,4) \to 1033, 193$
$(1,1,1,5) \to 901, 61$	$(1,1,2,5) \to 1027, 187$	$(1,1,3,5) \to 1153, 103$

Voici les unités de $\mathbb{Z}/104\mathbb{Z}$ au nombre de 48 : $\{1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 93, 95, 97, 99, 101, 103\}$.

L'expression mathématique qui représente le nombre de solutions envisageables S est :

$$S = \frac{\displaystyle\prod_{\substack{p_k \text{ premier} \\ 2 \leq p_k \leq \sqrt{n} \\ p_k | n}} (p_k - 1) \ . \prod_{\substack{p_{k'} \text{ premier} \\ 3 \leq p_{k'} \leq \sqrt{n} \\ p_{k'} \nmid n}} (p_{k'} - 2)}{\varphi(\displaystyle\prod_{\substack{p_{k''} \text{ premier} \\ 2 \leq p_{k''} \leq \sqrt{n}}} p_{k''})}$$

Puisque

$$\begin{aligned} \varphi(\prod_{\substack{p_k \text{ premier} \\ 2 \leq p_k \leq \sqrt{n}}} p_k) &= \prod_{\substack{p_k \text{ premier} \\ 2 \leq p_k \leq \sqrt{n}}} p_k \prod_{\substack{p_k \text{ premier} \\ 2 \leq p_k \leq \sqrt{n}}} \left(1 - \frac{1}{p_k}\right) \\ &= \prod_{\substack{p_k \text{ premier} \\ 2 \leq p_k \leq \sqrt{n}}} p_k \prod_{\substack{p_k \text{ premier} \\ 2 \leq p_k \leq \sqrt{n}}} \left(\frac{p_k - 1}{p_k}\right) \\ &= \prod_{\substack{p_k \text{ premier} \\ 2 \leq p_k \leq \sqrt{n}}} (p_k - 1), \end{aligned}$$

on trouve pour S l'expression

$$S = \prod_{\substack{p_k \text{ premier} \\ 3 \leq p_k \leq \sqrt{n} \\ p_k \nmid n}} \frac{p_k - 2}{p_k - 1}$$

Si les solutions envisageables sont équiréparties (on ne sait pas démontrer que c'est le cas), alors pour trouver celles qui sont comprises entre 3 et $n/2$, multiplier S par le ratio $\dfrac{n/2 - 2}{\displaystyle\prod_{\substack{p_k \text{ premier} \\ 2 \leq p_k \leq \sqrt{n}}} p_k}$ devrait compter les décompositions de Goldbach de n. Malheureusement, pour ce dernier calcul, le dénominateur augmentant beaucoup plus vite que le numérateur, on obtient un nombre de solutions de plus en plus petit. On ne sait pas pourquoi il est toujours possible de trouver une solution suffisamment petite (inférieure à $n/2$).

Références

[1] Donald Knuth, The Art of Computer Programming, vol. 1, Fundamental algorithms, Third edition, 1997 (pages 13 à 18).

[2] Denise Vella-Chemla, http://denise.vella.chemla.free.fr/jade1.pdf .

[3] Denise Vella-Chemla, http://denise.vella.chemla.free.fr/invariante.pdf .

[4] Denise Vella-Chemla, http://denise.vella.chemla.free.fr/DLpourCG.pdf .

Deux en moins (Denise Vella-Chemla, juillet 2022).

À cause d'une proposition de trouver les décomposants de Goldbach d'un nombre pair, supérieurs à la racine carrée de ce nombre pair, en éliminant une ou deux (soit systématiquement moins de 2) classes de congruences, selon tout module premier inférieur à la racine carrée du nombre pair considéré[1], on cherche à calculer $\prod_{2<p\leqslant n}\left(1-\frac{2}{p}\right)$.

On trouve une réponse dans un blog, postée par le pseudo abiessu vers 2011, modifiée vers 2018[2], dans laquelle est écrit :

"J'aimerais calculer, ou trouver une estimation raisonnable pour le produit ressemblant à un produit de Mertens suivant :

$$\prod_{2<p\leqslant n}\left(1-\frac{2}{p}\right)^{-1}=\prod_{2<p\leqslant n}\frac{p}{p-2}.$$

Faisons cela aussi explicitement que possible (je voudrais trouver la constante et le terme d'erreur).

1. Le problème original. D'abord, considérons l'identité :

$$\log\left(1-\frac{2}{p}+\frac{1}{p^2}\right)+\log\left(1-\frac{1}{(p-1)^2}\right)=\log\left(1-\frac{2}{p}\right).$$

De cela, il découle que :

$$\sum_{2<p\leqslant n}\log\left(1-\frac{2}{p}+\frac{1}{p^2}\right)+\sum_{2<p\leqslant n}\log\left(1-\frac{1}{(p-1)^2}\right)=\sum_{2<p\leqslant n}\log\left(1-\frac{2}{p}\right).$$

En multipliant par -1 et en prenant l'exponentielle des deux côtés amène :

$$\left(\prod_{2<p\leqslant n}1-\frac{1}{p}\right)^{-2}\cdot\prod_{2<p<n}\left(1-\frac{1}{(p-1)^2}\right)^{-1}=\left(\prod_{2<p\leqslant n}1-\frac{2}{p}\right)^{-1}$$

Rappelons que $\Pi_2=\prod_{2<p}\left(1-\frac{1}{(p-1)^2}\right)$ est la constante de Brun des nombres premiers jumeaux, et que le produit du côté gauche converge vers l'inverse de celle-ci. C'est alors par une formule de Mertens que l'on sait que :

$$\left(\prod_{2<p\leqslant n}1-\frac{1}{p}\right)^{-1}=\frac{1}{2}e^{\gamma}\log n+O(1)$$

où γ est la constante d'Euler-Mascheroni (c'est précisément le théorème 2.7 (e) dans le livre de Montgomery *Multiplicative Number Theory I. Classical Theory*). Par mise au carré de ce résultat asymptotique, on peut conclure :

$$\left(\prod_{2<p\leqslant n}1-\frac{2}{p}\right)^{-1}=\frac{1}{4}e^{2\gamma}\Pi_2^{-1}\log^2 n+O(\log n)$$

[1]://denise.vella.chemla.free.fr/jade1.pdf.
[2]https://math.stackexchange.com/questions/22411/computing-the-product-of-p-p-2-over-the-odd-primes?rq=1

Espérons que ça aide.

Note : j'ai substitué par la constante des nombres premiers jumeaux pour cet autre produit parce qu'elle converge très rapidement en comparaison avec le terme d'erreur. Je peux donner plus de détails si souhaité, mais je laisse cela en exercice.

2. Est-il possible de faire mieux ? Le terme d'erreur peut-il être amélioré ? Oui. Il s'avère que l'on peut rendre le terme d'erreur bien meilleur. En utilisant le théorème des nombres premiers, on trouve,

$$\left(\prod_{2<p\leqslant n} 1-\frac{1}{p}\right)^{-1} = \frac{1}{2}e^{\gamma}\log n + e^{-c\sqrt{\log n}}$$

où c est la constante utilisée dans la preuve de la région sans zéro[3]. Puisque $e^{-\sqrt{\log n}}$ décroît plus rapidement que n'importe quelle puissance de log, on obtient un bien meilleur résultat en élevant au carré cette estimation. Précisément, on a :

$$\left(\prod_{2<p\leqslant n} 1-\frac{2}{p}\right)^{-1} = \frac{1}{4}e^{2\gamma}\Pi_2^{-1}\log^2 n + O(e^{-c\sqrt{\log n}})$$

(à nouveau, la convergence vers Π_2 est beaucoup trop rapide pour interférer avec le terme d'erreur).

Je souhaiterais parier que c'est le mieux que l'on puisse faire, et que quel que soit ce que l'on pourrait faire de mieux, cela impliquerait des résultats plus forts sur le terme d'erreur pour $\pi(x)$, la fonction de comptage des nombres premiers.

3. Résultats numériques. Juste pour le plaisir, la constante exacte, devant le terme $\log^2 n$ est : 1.201303 [4]. C'est proche comment ? Bien, pour :

$n = 10$
on obtient une erreur de 0.630811.

$n = 50$
on obtient une erreur de 1.22144

$n = 100$
on obtient une erreur de 0.63493

$n = 1000$
on obtient une erreur de 0.438602

[3] Problème : avec les valeurs fournies dans la section 3 ci-après, je ne peux trouver quelle constante a été utilisée pour calculer le terme d'erreur.

[4] On rappelle les valeurs des différentes constantes : avec $e = 2.71828, \gamma = 0.5772156649$, on a $e^{2\gamma} = 3.17221895813$ dont le quart vaut 0.79305473953. Sur la toile, la constante de Brun est indiquée comme valant $\Pi_2 = 0.6601618158$ (voir https://mathworld.wolfram.com/TwinPrimesConstant.html), dont l'inverse est 1.51478012825, qui multiplié par le reste donne bien la valeur fournie de 1.201303 comme constante à multiplier par $\log^2 n$ dans la dernière formule ci-dessus.

$n = 10^4$
on obtient une erreur de 0.250181

$n = 10^5$
on obtient une erreur de 0.096783

$n = 10^6$
on obtient une erreur de 0.017807

où chaque fois, l'erreur est positive. C'est-à-dire que le produit semble être légèrement plus grand que l'asymptote (mais converger assez rapidement). Pourtant, mon intuition me dit qu'il est presque certain (je ne le prouverai pas ici) que le terme d'erreur oscille de négatif à positif infiniment souvent.

4. Sous l'hypothèse de Riemann.
Si l'on suppose que l'hypothèse de Riemann est vérifiée, le terme d'erreur est borné par

$$\frac{C \log^2 x}{\sqrt{x}}$$

pour une certaine constante C. En analysant les données ci-dessus avec des méthodes numériques, le terme d'erreur $\frac{C \log x}{\sqrt{x}}$ semble être mieux adapté".

Annexe : Traductions de deux extraits de *Multiplicative Number Theory : I. Classical theory* de H. L. Montgomery et R. C. Vaughan (p. 46-48)

(p. 13)

Théorème 1.3. *Soit* $A(x) = \sum_{n \leq x} a_n$. *Si* $\sigma_c < 0$, *alors* $A(x)$ *est une fonction bornée et*

$$\sum_{n=1}^{\infty} a_n n^{-s} = s \int_1^{\infty} A(x) x^{-s-1} dx \tag{1.10}$$

pour $\sigma > 0$. *Si* $\sigma_c \geq 0$, *alors*

$$\limsup_{x \to \infty} \frac{\log |A(x)|}{\log x} = \sigma_c, \tag{1.11}$$

et (1.10) et vérifiée pour $\sigma > \sigma_c$.

(p. 22-23)

$$\sum_{d|n} \mu(d) = \begin{cases} 1 & \text{si} \quad n = 1 \\ 0 & \text{si} \quad n > 1. \end{cases} \tag{1.20}$$

(p. 23)

Corollaire 1.11. *Pour* $\sigma > 1$
$$\log\zeta(s) = \sum_{n=1}^{\infty} \frac{\Lambda(n)}{\log\, n} n^{-s}$$
et
$$-\frac{\zeta'(s)}{\zeta(s)} = \sum_{n=1}^{\infty} \Lambda(n) n^{-s}.$$

(p. 24)

$$\sum_{d|n} \Lambda(d) = \log\, n. \tag{1.22}$$

(p. 25)

Corollaire 1.13. *La fonction zeta de Riemann a un pôle simple en* $s = 1$*, mais sinon est analytique dans le demi-plan* $\sigma > 0$*.*

(p. 35)

2.1. Valeurs moyennes

On dit qu'une fonction arithmétique $F(n)$ a une valeur moyenne c si
$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} F(n) = c.$$

Dans cette section, on développe une méthode simple par laquelle on peut montrer l'existence de valeurs moyennes dans de nombreux cas intéressants.

Si deux fonctions arithmétiques f et F sont reliées par l'identité
$$F(n) = \sum_{d|n} f(d), \tag{2.1}$$
alors on peut écrire f en fonction de F :
$$f(n) = \sum_{d|n} \mu(d) F(n/d). \tag{2.2}$$

Cette formule est appelée *Formule d'inversion de Möbius*. Inversement, si (2.2) est vérifiée pour tout n alors (2.1) l'est également. Si f est généralement petit alors F a une valeur moyenne asymptotique. Pour voir cela, observons que
$$\sum_{n\le x} F(n) = \sum_{n\le x} \sum_{d|n} f(d).$$

En itérant les sommes dans l'ordre inverse, on voit que l'expression ci-dessus est

$$= \sum_{d \le x} f(d) \sum_{\substack{n \le x \\ d|n}} 1 = \sum_{n \le x} f(d)[x/d].$$

Puisque $[y] = y + O(1)$, ceci est

$$= x \sum_{d \le x} \frac{f(d)}{d} + O\left(\sum_{d \le x} |f(d)|\right). \tag{2.3}$$

Ainsi F a la valeur moyenne $\sum_{d=1}^{\infty} f(d)/d$ si cette série converge et si $\sum_{d \le x} |f(d)| = o(x)$. Cette approche, quoique assez brutale, amène souvent des résultats utiles.

---

p. 37.

Supposons que a_k, b_m, c_n soient liés par la relation de convolution

$$c_n = \sum_{km=n} a_k b_m, \tag{2.5}$$

et que $A(x), B(x), C(x)$ soient leurs fonctions de sommation respectives. Alors

$$C(x) = \sum_{km=n} a_k b_m, \tag{2.6}$$

et il est utile de noter que cette double somme peut être itérée de différentes manières. D'un côté, on voit que

$$C(x) = \sum_{k \le x} a_k B(x/k)\ ; \tag{2.7}$$

(p. 46)

2.2. Les estimations du nombre de nombres premiers de Chebyshev et de Mertens

À cause de l'espacement irrégulier entre les nombres premiers, il semble sans espoir de donner une formule exacte pour le $n^{\text{ième}}$ nombre premier. Comme compromis, on estime le $n^{\text{ième}}$ nombre premier, ou de façon équivalente on estime le nombre $\pi(x)$ de nombres premiers n'excédant pas x. De façon similaire, on pose $\vartheta(x) = \sum_{p \le x} \log\, p$, and $\psi(x) = \sum_{n \le x} \Lambda(n)$. Comme nous le verrons, ces trois fonctions sommatoires sont intimement liées. Estimons d'abord $\psi(x)$.

Théorème 2.4 (Chebyshev) *Pour $x \ge 2, \psi(x) \asymp x$.*

La preuve que nous donnons ci-dessous établit seulement qu'il existe un certain x_0 tel que $\psi(x) \asymp x$ uniformément pour $x \ge x_0$. Pourtant, à la fois $\psi(x)$ et x sont bornées en dehors de 0 et de ∞ dans l'intervalle $[2, x_0]$, et par conséquent les constantes implicites peuvent être ajustées de telle façon que l'on ait $\psi(x) \asymp x$ pour $x \ge 2$. Dans les situations ultérieures de cette sorte, on supposera sans commentaire que le lecteur comprend qu'il suffit de démontrer le résultat pour tout x suffisamment grand.

Preuve. En appliquant la formule d'inversion de Möbius à (1.22), on trouve que

$$\Lambda(n) = \sum_{d|n} \mu(d)\, \log\, n/d.$$

Ainsi, de (2.7), il découle que

$$\psi(x) = \sum_{d\le x} \mu(d)T(x/d) \tag{2.10}$$

où $T(x) = \sum_{n\le x} \log n$. Par le test de l'intégrale, on voit que

$$\int_1^N \log u \, du \le T(N) \le \int_1^{N+1} \log u \, du$$

pour tout entier positif N. Puisque $\int \log x \, dx = x \log x - x$, il s'ensuit facilement que

$$T(x) = x \log x - x + O(\log 2x) \tag{2.11}$$

pour $x \ge 1$. Malgré la précision de cette estimation, on rencontre des difficultés quand on substitue cela dans (2.10), puisqu'on n'a pas d'information utile concernant les sommes

$$\sum_{d\le x} \frac{\mu(d)}{d}, \qquad \sum_{d\le x} \frac{\mu(d)\log d}{d},$$

qui émergent dans les termes principaux. Pour éviter ce problème, on introduit une idée qui est fondamentale dans de grands pans de la théorie des nombres premiers, notamment, on remplace $\mu(d)$ par une fonction arithmétique a_d qui d'une certaine manière forme une approximation tronquée de $\mu(d)$. Supposons que $\mathcal{D}$ est un ensemble fini de nombres, et que $\mathrm{a}_d = 0$ quand $d \notin \mathcal{D}$. Alors par (2.11), on voit que

$$\sum_{d\in\mathcal{D}} a_d T(x/d) = (x \log x - x) \sum_{d\in\mathcal{D}} a_d/d - x \sum_{d\in\mathcal{D}} \frac{a_d \log d}{d} + O(\log 2x). \tag{2.12}$$

Ici, la constante implicite dépend du choix de a_d, que nous considérerons comme étant fixé. Puisque nous voulons que la relation ci-dessus approxime la relation (2.10), et puisque nous espérons que $\psi(x) \asymp x$, nous restreignons notre attention aux a_d qui satisfont la condition

$$\sum_{d\in\mathcal{D}} \frac{a_d}{d} = 0, \tag{2.13}$$

et espérer que

$$-\sum_{d\in\mathcal{D}} \frac{a_d \log d}{d} \text{ est proche de 1.} \tag{2.14}$$

Par la définition de $T(x)$, on voit que le côté gauche de (2.12) est

$$\begin{aligned} \sum_{dn\le x} a_d \log n &= \sum_{dn\le x} a_d \sum_{k|n} \Lambda(k) = \sum_{dkm\le x} a_d \Lambda(k) \\ &= \sum_{k\le x} \Lambda(k) E(x/k) \end{aligned} \tag{2.15}$$

où $E(y) = \sum_{dm\le y} a_d = \sum_d a_d [y/d]$. L'expression ci-dessus sera proche de $\psi(x)$ si $E(y)$ est proche de 1. Si $y \ge 1$ alors

$$\sum_d \mu(d)[y/d] = \sum_d \mu(d) \sum_{k\le y/d} 1 = \sum_{dk\le y} \mu(d) = \sum_{n\le y} \sum_{d|n} \mu(d) = 1,$$

en voyant (1.20). Ainsi $E(y)$ sera proche de 1 pour y pas trop grand si a_d est proche est $\mu(d)$ pour de petites valeurs de d. De plus, par (2.13), on voit que $E(y) = -\sum_{d\in\mathcal{D}} a_d[y/d]$, de telle façon que

$E(y)$ est périodique avec une période qui divise $\mathrm{lcm}_{d\in\mathcal{D}}\, d$. Par conséquent, pour un choix donné de a_d, le comportement de $E(y)$ peut être déterminé par un calcul fini.

La réalisation la plus simple de cette approche implique de prendre $a_1 = 1, a_2 = -2, a_d = 0$ pour $d > 2$. Alors (2.13) est vérifiée, l'expression (2.14) est log 2, $E(y)$ a pour période 2 et $E(y) = 0$ pour $0 \leq y < 1$, $E(y) = 1$ pour $1 \leq y < 2$. Par conséquent, pour ce choix de a_d, la somme en (2.15) satisfait les inégalités

$$\psi(x) - \psi(x/2) = \sum_{x/2<k\leq x} \Lambda(k) \leq \sum_{k\leq x} \Lambda(k)E(x/k) \leq \sum_{k\leq x} \Lambda(k) = \psi(x)$$

Donc $\psi(x) \geq (\log 2)x + O(\log x)$, qui est une borne inférieure de la forme désirée. De plus,

$$\psi(x) - \psi(x/2) \leq (\log 2)x + O(\log x).$$

En remplaçant x par $x/2^r$ et en sommant sur r, on déduit que

$$\psi(x) \leq 2(\log 2)x + O((\log x)^2),$$

ainsi, la démonstration est terminée. $\square$

(p. 50-52)

Théorème 2.7 Pour $x \geq 2$,

(a) $\displaystyle\sum_{n\leq x} \frac{\Lambda(n)}{n} = \log x + O(1)$,

(b) $\displaystyle\sum_{p\leq x} \frac{\log p}{p} = \log x + O(1)$,

(c) $\displaystyle\int_1^x \psi(u)u^{-2}du = \log x + O(1)$,

(d) $\displaystyle\sum_{p\leq x} \frac{1}{p} = \log\log x + b + O(1/\log x)$,

(e) $\displaystyle\prod_{p\leq x} \left(1 - \frac{1}{p}\right)^{-1} = e^{C_0}\log x + O(1)$
où C_0 est la constante d'Euler et

$$b = C_0 - \sum_p \sum_{k=2}^{\infty} \frac{1}{kp^k}.$$

Preuve. En prenant $f(d) = \Lambda(d)$ dans (2.1), on voit grâce à (2.3) que

$$T(x) = \sum_{n\leq x} \log n = x \sum_{n\leq x} \frac{\Lambda(d)}{d} + O(\psi(x)).$$

Par le théorème 2.4, le terme d'erreur est $\ll x$. Ainsi (2.11) donne (a). La somme en (b) diffère de celle en (a) par ceci

$$\sum_{\substack{p^k \leq x \\ k \geq 2}} \frac{\log p}{p^k} \leq \sum_p \frac{\log p}{p(p-1)} \ll 1.$$

Pour déduire (c), on note que la somme en (a) est

$$\int_{2^-}^{x} u^{-1} d\psi(u) = \frac{\psi(u)}{u}\Big|_2^{x^-} + \int_2^x \psi(u)u^{-2}du = \int_2^x \psi(u)u^{-2}du + O(1)$$

par le théorème 2.4. On démontre maintenant (d) sans déterminer la valeur de la constante b. On exprime (b) sous la forme $L(x) = \log x + R(x)$ où $R(x) \ll 1$. Alors

$$\begin{aligned}
\sum_{p \leq x} \frac{1}{p} &= \int_{2^-}^{x} (\log u)^{-1} dL(u) = \int_{2^-}^{x} \frac{1}{\log u} d\log u + \int_{2^-}^{x} \frac{d\,R(u)}{\log u} \\
&= \int_{2^-}^{x} \frac{du}{u \log u} + \left[\frac{R(u)}{\log u}\Big|_{2^-}^{x} - \int_{2^-}^{x} R(u) d(\log u)^{-1}\right. \\
&= \log\log x - \log\log 2 + 1 + \frac{R(x)}{\log x} + \int_2^x \frac{R(u)}{u(\log u)^2}\,du.
\end{aligned}$$

Le dernier terme est $\ll 1/\log x$, et l'intégrale est $\int_2^\infty - \int_x^\infty = \int_2^\infty + 0(1/\log x)$, de telle façon qu'on a (d) avec

$$b = 1 - \log\log 2 + \int_2^\infty \frac{R(u)}{u(\log u)^2} du.$$

Comme pour (e), on note que

$$\sum_{p \leq x} \log\left(1 - \frac{1}{p}\right)^{-1} = \sum_{p \leq x} \frac{1}{p} + \sum_{p \leq x} \left(\log\left(1 - \frac{1}{p}\right)^{-1} - \frac{1}{p}\right).$$

La seconde somme sur la droite est

$$\sum_p \sum_{k=2}^{\infty} \frac{1}{kp^k} + O\left(\sum_{p > x} p^{-2}\right)$$

et le terme d'erreur est ici $\ll \sum_{n>x} n^{-2} \ll x^{-1}$, de telle façon que de (d), on a

$$\sum_{p \leq x} \log\left(1 - \frac{1}{p}\right)^{-1} = \log\log x + c + O(1/\log x) \tag{2.16}$$

où $c = b + \sum_p \sum_{k \geq 2} (kp^k)^{-1}$. Puisque $e^z = 1 + O(|z|)$ pour $|z| \leq 1$, en prenant l'exponentielle, on déduit que

$$\prod_{p \leq x} \left(1 - \frac{1}{p}\right)^{-1} = e^c \log x + O(1).$$

Pour compléter la démonstration, il suffit de montrer que $c = C_0$. Pour cela, on note d'abord que si $p \le x$ et $p^k > x$, alors $k \ge (\log x)/\log p$. Par conséquent

$$\sum_{\substack{p \le x \\ p^k > x}} \frac{1}{kp^k} \ll \sum_{\substack{p \le x \\ p^k > x}} \frac{\log p}{(\log x)\, p^k} \ll \sum_p \frac{\log p}{\log x} \sum_{k \ge 2} p^{-k} \ll \frac{1}{\log x} \sum_p \frac{\log p}{p^2} \ll \frac{1}{\log x},$$

de telle façon qu'à partir de (2.16) on a

$$\sum_{1 < n \le x} \frac{\Lambda(n)}{n \log n} = \log \log x + c + O(1/\log x).$$

Par le corollaire 1.15, on peut réécrire cela en

$$\sum_{1 < n \le x} \frac{\Lambda(n)}{n \log n} = \sum_{n \le \log x} \frac{1}{n} + (c - C_0) + O(1/\log 2x).$$

Puisque ceci est trivial quand $1 \le x < 2$, la formule ci-dessus est vérifiée pour tout $x \ge 1$. On exprime cela brièvement par $T_1 = T_2 + T_3 + T_4$, et on estime les quantités $I_i = \delta \int_1^\infty x^{-1-\delta} T_i(x) dx$. En comparant les résultats lorsque $\delta \to 0^+$, on déduira que $c = C_0$. Par le théorème 1.3, corollaire 1.11, et corollaire 1.13, on voit que

$$I_1 = \log \zeta(1+\delta) = \log \frac{1}{\delta} + O(\delta)$$

lorsque $\delta \to 0^+$. Deuxièmement,

$$\begin{aligned} I_2 &= \delta \sum_{n=1}^\infty \frac{1}{n} \int_{e^n}^\infty x^{-1-\delta} dx = \sum_{n=1}^\infty \frac{1}{n} e^{-\delta n} = \log(1 - e^{-\delta})^{-1} \\ &= \log(\delta + O(\delta^2))^{-1} = \log 1/\delta + O(\delta). \end{aligned}$$

Troisièmement

$$I_3 = c - C_0,$$

et finalement

$$I_4 \ll \delta \int_1^\infty x^{-1-\delta} \frac{dx}{\log 2x} \ll \delta + \delta \int_2^{e^{1/\delta}} \frac{dx}{x \log x} + \delta^2 \int_{e^{1/\delta}}^\infty x^{-1-\delta} dx \ll \delta \log 1/\delta.$$

Puisque les principaux termes s'éliminent, en faisant tendre $\delta \to 0^+$, on voit que $c = C_0$.

$\square$

99,718 % n'est pas 100 % (Denise Vella-Chemla, 22.07.2022)

1. Une fonction polynomiale du 3ème degré.

Dans une note précédente[1], on a démontré qu'un nombre premier supérieur à $\sqrt{n}$ et inférieur ou égal à $n/2$, et qui n'est jamais congru à n modulo tout nombre premier inférieur à $\sqrt{n}$ est un décomposant de Goldbach de n.

Il s'agit de démontrer qu'un tel nombre existe toujours et on a essayé d'étudier si des résultats d'analyse permettraient de savoir si l'existence d'un décomposant de Goldbach est toujours garantie, lorsqu'on élimine au maximum 2 classes de congruence par module [2].

Ici, on teste une tout autre idée, algébrique : on cherche une fonction polynomiale $x \mapsto f(x)$, qu'on applique aux nombres impairs de 3 à $n/2$ pour n un nombre pair. On souhaiterait que cette fonction rende fixe un décomposant de Goldbach au moins, ou bien qu'elle envoie un décomposant de Goldbach sur son complémentaire à n, tout en envoyant les non décomposants de Goldbach sur d'autres nombres que leur complémentaire à n ; ou bien, on teste si la fonction polynomiale considérée envoie un certain décomposant de Goldbach de n sur un autre décomposant de Goldbach de n, tout ceci ayant lieu dans l'anneau $\mathbb{Z}/n\mathbb{Z}$.

On a trouvé une telle application polynomiale "sympathique" : $x \mapsto x^3 - x^2 + 1$. On considère que l'image de p est "satisfaisante" si et seulement si on a à la fois p premier et $n-p$ premier et $f(p)$ premier et $f(n-p)$ premier (en acceptant que $f(p)$ puisse être égal à p ou bien à $n-p$).

Mais pour 10 % des nombres jusqu'à 100, aucun nombre entier impair de $[3, n/2]$ n'a une image satisfaisante. Ce qui est un pourcentage conséquent.

Pour 9.03 % des nombres jusqu'à 1000 (resp. 2,60 % des nombres jusqu'à 10^4), même chose.

Ce ratio décroît bien à 0,282 % seulement des nombres jusqu'à 100 000, pour lesquels l'application polynomiale ne permet pas de trouver un décomposant de Goldbach de n. Jusqu'à 10^5, on parvient à trouver (par programme) au moins un décomposant de Goldbach dont l'image par l'application est satisfaisante dans 99,718 % des nombres pairs.

On essaie de caractériser les nombres qui posent problème pour voir s'ils ont des propriétés particulières (en terme de factorisation par exemple), on n'y parvient pas[3]. De plus, ça semble aussi difficile de démontrer qu'un nombre premier a pour image un nombre premier par cette application

[1] http://denise.vella.chemla.free.fr/jade1.pdf.

[2] Puisqu'on peut trouver un décomposant de Goldbach en criblant (éliminant) au maximum deux classes de congruences selon tout nombre premier p inférieur à $\sqrt{n}$, on peut essayer d'utiliser le produit $\displaystyle\prod_{\substack{p \text{ premier} \\ p \leq \sqrt{n}}} \left(1 - \frac{2}{p}\right)$. Voir la note http://denise.vella.chemla.free.fr/deux-en-moins.pdf.

[3] Voici les factorisations de certains nombres pour lesquels l'application polynomiale n'amène à rien : $\{4{\cdot}8641, 2 \cdot 41 \cdot 419, 2^2 \cdot 6961, 2^6 \cdot 311, 2\dot{3}7 \cdot 227, 2^2 \cdot 3533, 2 \cdot 41 \cdot 163, 2^3 \cdot 1627, 2^4 \cdot 797, 2 \cdot 31 \cdot 197, \textit{etc.}\}$.

que de démontrer la conjecture sans en passer par ce détour.

2. Une fonction polynomiale du 2^d degré.

On a également fait des tests avec cette autre fonction :

$$x \mapsto f(x) = \begin{cases} -x^2 + \left(\dfrac{n}{2} - 1\right) x & \text{si } x \text{ est de la forme } 4k+2; \\ \\ -x^2 + \dfrac{n}{2} x & \text{si } x \text{ est de la forme } 4k \end{cases}$$

Le rendement de cette fonction est moins bon. Les taux de ratage sont de 12,24 % jusqu'à 1000, 9,6718 % jusqu'à 10^3, 3,06 % jusqu'à 10^4 et enfin 0,33801 % jusqu'à 10^5 (> 0,282 % trouvé pour la fonction du troisième degré). Ces échecs sont problématiques de toute façon, de par leur nombre, et de par le fait qu'on n'arrive pas à comprendre dans quels cas particuliers ils se produisent.

Calcule et ... (Denise Vella-Chemla, juillet 2022)

On reprend notre exemple fétiche de la recherche des décomposants de Goldbach de l'entier pair $n = 98$.

$$S_{98} = \begin{cases} 98 \equiv 0 \pmod 2 \\ 98 \equiv 2 \pmod 3 \\ 98 \equiv 3 \pmod 5 \\ 98 \equiv 0 \pmod 7 \end{cases}$$

Appelons d_{98} un décomposant de Goldbach potentiel de $n = 98$. d_{98} peut être congru, hormis 0, à tout ce à quoi $n = 98$ n'est pas congru. Le signe $\vee$ dans le système ci-dessous est à lire comme un ou exclusif, son emploi étendu est à comprendre comme le fait de vérifier autant de systèmes de congruences que la combinatoire le permet.

$$S_{d_{98}} = \begin{cases} d_{98} \equiv 1 \pmod 2 \\ d_{98} \equiv 1 \pmod 3 \\ d_{98} \equiv 1 \vee 2 \vee 4 \pmod 5 \\ d_{98} \equiv 1 \vee 2 \vee 3 \vee 4 \vee 5 \vee 6 \pmod 7 \end{cases}$$

Remarque : on notera que le fait de respecter le système de systèmes de congruences ci-dessus est une condition suffisante mais non nécessaire pour être un décomposant de Goldbach de n. On trouvera la preuve de cette caractérisation des décomposants de Goldbach d'un nombre pair n qui sont supérieurs à la racine carrée de n en [1].

Comme on peut le comprendre, les modules qui ne divisent pas n "éliminent davantage de classes de congruences" (au nombre de 2) que les modules qui divisent n. Plaçons-nous dans le pire des cas, où l'on élimine deux classes de congruences par module premier inférieur à $\sqrt{n}$, on trouve tout de même

$$\frac{1}{2} \prod_{\substack{p \text{ premier} \\ 5 \leq p \leq \sqrt{n}}} (p-2)$$

classes de congruences différentes par l'application du théorème des restes chinois à chacun des systèmes de congruences combinatoirement trouvé (voir $S_{d_{98}}$ ci-dessus). La division par 2 est justifiée par les symétries autour des moitiés (par exemple, pour 40, les classes de congruences trouvées par l'application du théorème des restes chinois à chaque système[1] sont les classes $30k+11, 30k+17, 30k+23$ et $30k+29$ dont on ne conserve que la moitié par symétrie autour de 20, la moitié de 40.

Mais d'autre part, les solutions étant toutes des unités du groupe $(\mathbb{Z}/n\mathbb{Z})^{\times}$, la moitié d'entre elles sont inférieures à $D = \dfrac{1}{2} \displaystyle\prod_{\substack{p \text{ premier} \\ 3 \leq p \leq \sqrt{n}}} p$ (pour illustrer cela sur l'exemple $n = 98$, la moitié des solutions (s'il en existe) sont forcément inférieures à 105=3×5 × 7).

[1]Systèmes de congruences pour $n = 40$:

$$\begin{cases} 40 \equiv 0 \pmod 2 \\ 40 \equiv 1 \pmod 3 \\ 40 \equiv 0 \pmod 5 \end{cases} \qquad S_{d_{40}} = \begin{cases} d_{40} \equiv 1 \pmod 2 \\ d_{40} \equiv 2 \pmod 3 \\ d_{40} \equiv 1 \vee 2 \vee 3 \vee 4 \pmod 5 \end{cases}$$

Serait-il possible de "rater l'intervalle visé", i.e. que toutes les solutions soient supérieures à n, comprises entre n et D ?

Imaginons qu'il existe un corps K dont on serait assuré qu'une de ses extensions E/K contienne l'une des solutions (on n'a pas trouvé d'équation polynomiale générale que seuls les décompoosants de Goldbach de n satisferaient). L'existence d'un décomposant de Goldbach pourrait découler d'une minoration trouvée dans [2] p. 7. Le théorème de densité assure l'existence d'un certain nombre de nombres premiers inférieurs à une certaine borne, et ce qui garantirait l'existence d'un décomposant de Goldbach serait que cette borne soit inférieure à n.

Le manque de technique en mathématiques est très handicapant.

Référence

[1] D. Chemla, *Réécrire*, http://denise.vella.chemla.free.fr/jade1.pdf

[2] J.-P. Serre, *Quelques applications du théorème de densité de Chebotarev*, Publications Mathématiques de l'IHÉS, Tome 54 (1981), pp. 123-201.

Annexe : extraits de [2] fournissant la minoration (p. 5 et 7)

§ 1. Majorations de discriminants

1.1. Notations

Soit K un corps de nombres algébriques, autrement dit une extension finie de $\mathbf{Q}$. On pose :

A_K anneau des entiers de K,
$n_K = [K : \mathbf{Q}] = [A_K : \mathbf{Z}] =$ degré de K,
$\sum_K$=ensemble des places ultramétriques de K,
$d_K =$ valeur absolue du discriminant de K.

Si $v \in \sum_K$, on identifie v à la valuation discrète normée correspondante (de groupe des valeurs $\mathbf{Z}$), et l'on note $\mathfrak{p}_v$, l'idéal premier de A_K qui correspond à v. Le corps résiduel $A_K/\mathfrak{p}_v$, est un corps fini; on note p_v sa caractéristique, et Nv le nombre de ses éléments. On a

$$Nv = N\mathfrak{p}_v = (p_v)^{f_v},$$

où f_v est le degré résiduel de v. L'indice de ramification e_v de v est défini par $e_v = v(p_v)$; c'est le plus grand entier positif e tel que $\mathfrak{p}_v^e$ divise p_v.

Soit E une extension finie de K, de degré $n = n_E/n_K = [E : K]$. On note $\mathfrak{D}_{E/K}$ (resp. $\mathfrak{d}_{E/k}$) la différente (resp. le discriminant) de l'extension E/K ; c'est un idéal $\neq 0$ de A_E (resp. A_K). On a

(1) $$\mathfrak{d}_{E/K} = N_{E/K}(\mathfrak{D}_{E/K}) \qquad \text{et} \qquad d_E = (d_K)^n N(\mathfrak{d}_{E/K}),$$

cf. par exemple [38], chap. III.

1.2. Estimations locales

Soit E/K comme ci-dessus, et soit $w \in \sum_E$ une place ultramétrique de E. Notons v la place de K induite par w, et soit $e_{w/v} = e_w/e_v$ l'indice de ramification de w par rapport à v. On s'intéresse à *l'exposant* $w(\mathfrak{D}_{E/K})$ de l'idéal premier $\mathfrak{p}_w$ dans la différente $\mathfrak{D}_{E/K}$:

Proposition 1. On a

$$w(\mathfrak{D}_{E/K}) = e_{w/v} - 1 + s_{w/v}, \qquad \textit{avec} \qquad 0 \leq s_{w/v} \leq w(e_{w/v}).$$

[...]

En appliquant la prop. 1, on en déduit

$$\begin{aligned} v(\mathfrak{d}_{E/K}) &= \sum_{w|v} f_{w/v}(e_{w/v} - 1) + \sum_{w|v} f_{w/v} s_{w/v} \\ &\leq \sum_{w|v} f_{w/v} e_{w/v} - 1 + \sum_{w|v} f_{w/v} e_w v_p(e_{w/v}) \\ &\leq n - 1 + n e_v \operatorname*{Sup}_{w|v} v_p(e_{w/v}), \end{aligned} \tag{2}$$

du fait que $n = \sum_{w|v} f_{w/v} e_{w/v}$.

[...]

D'autre part, comme E/K est ramifiée en v, on a $e \geq 2$, et la formule (2) ci-dessus montre que

$$v(\mathfrak{d}_{E/K}) \geq gf(e-1) \geq n(1 - 1/e) \geq n/2.$$

Raison de la curiosité pour l'équation de Chazy (Denise Vella-Chemla, juillet 2022)

J'aimerais comprendre cette équation pour la raison suivante : au début de mes recherches, j'ai été fascinée par l'article d'Euler *Découverte d'une loi tout extraordinaire des nombres par rapport à la somme de leurs diviseurs*, un des rares articles qu'Euler ait écrit en français.

Un nombre premier se distingue d'un nombre composé par le fait que sa somme des diviseurs est le nombre qui le suit ($\sigma(p) = p + 1$ si et seulement si p est un nombre premier).

Je voyais donc initialement au-dessus de mon maillage[1] une surface bosselée dont les minimums locaux sur des diagonales descendantes seraient comme des "trous Goldbach" qui minimiseraient la fonction $\sigma(p) + \sigma(n - p)$.

J'avais demandé à Jean-François Colonna de dessiner une telle surface mais le résultat n'a pas été probant[2].

J'ai programmé la récurrence fournie par Euler mais les conditions de calcul dans le cas limite était un peu bizarres, puis j'ai trouvé cette récurrence très simple sur la toile :

$$\begin{cases} \sigma(1) = 1 \\ \sigma(n) = \dfrac{12}{n^3 - n^2} \displaystyle\sum_{k=1}^{n-1} (-5k^2 + 5kn - n^2)\sigma(k)\sigma(n-k). \end{cases}$$

On peut remplacer le $-5k^2 + 5kn - n^2$ par $k(3n - 5k)$ et ça semble fonctionner aussi.

En fait, le lien avec l'équation de Chazy est établi par un passage par la série $E_2(t)$ d'Eisenstein

$$y(t) = i\pi E_2(t) = i\pi \left(1 - 24 \sum_{n=1}^{\infty} \sigma(n) q^n\right)$$

avec $q = e^{2\pi i t}$ et la fonction y qui vérifie l'équation de Chazy $y''' - 2yy'' + 3y'^2 = 0$.

C'est pour cette raison que j'ai traduit le texte de Clarkson et Olver : je suis bloquée par l'anglais *et* par la technique mathématique. Lire le texte en français atténue un peu l'une des deux difficultés (si ce n'est qu'il ne faudrait pas que des erreurs de traduction annule tout le bénéfice acquis par ladite traduction).

Le but idéal serait de trouver une expression toute simple pour distinguer les nombres premiers des nombres composés.

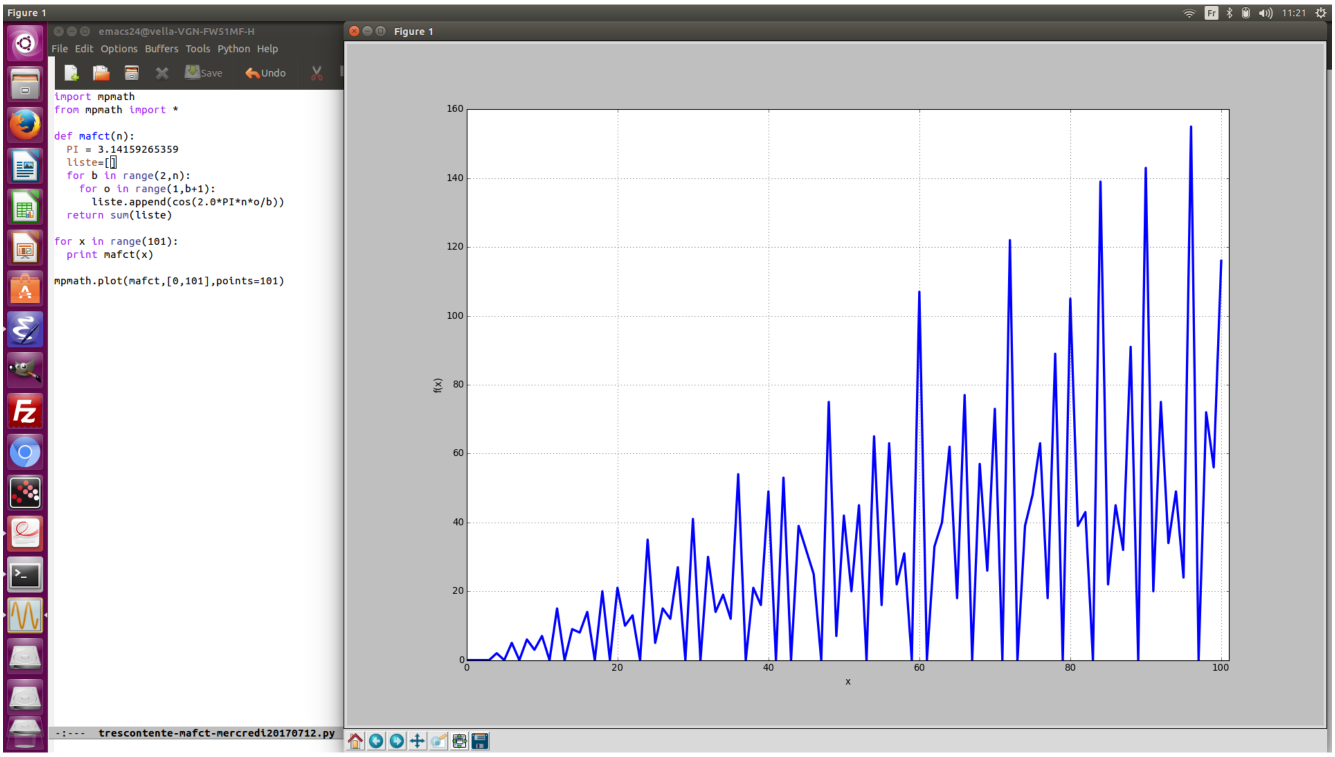
[1] Voir http://denise.vella.chemla.free.fr/treillisnp.html.

[2] Voir https://images.math.cnrs.fr/spip.php?page=imageid$_d ocument = 19562 lang = es$ et http://denisevellachemla.eu/extraits-Colonna.pdf.

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, $\sigma(p) = p + 1$[†]) :

$$n \; est \; premier \iff \sum_{b=2}^{n-1} \sum_{o=1}^{b} cos \frac{2\pi n o}{b} = 0$$



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)
```

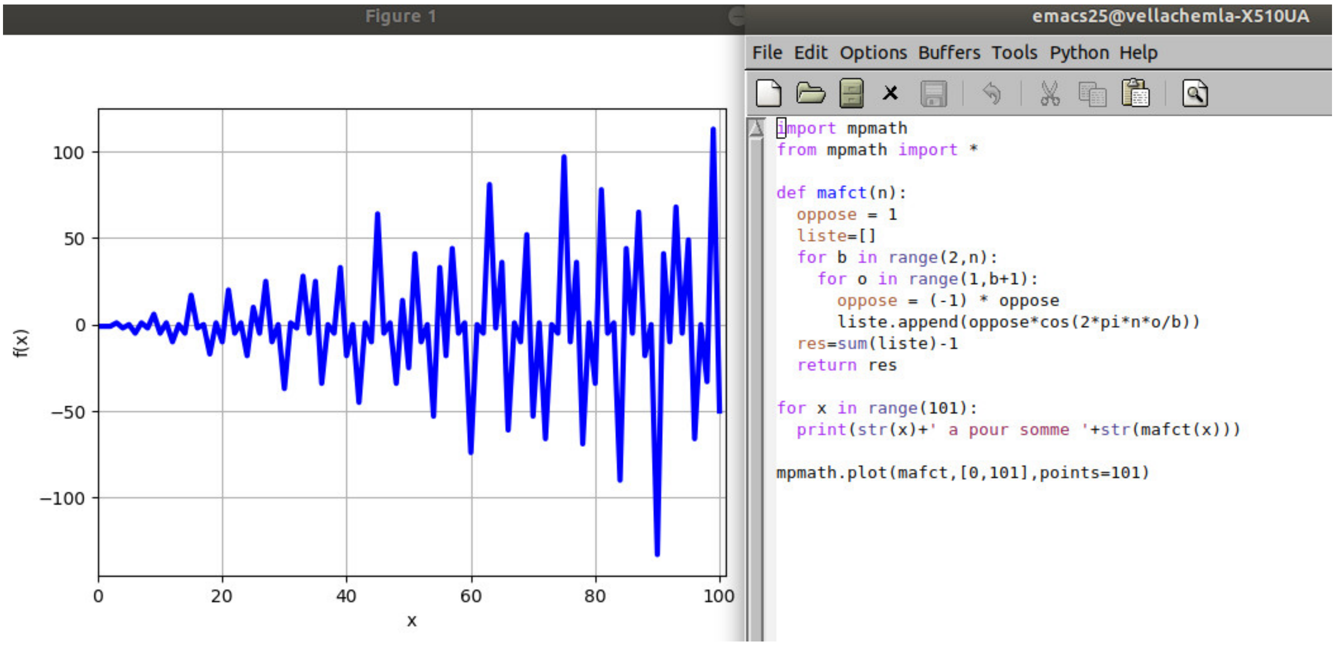


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
100
50
0
−50
−100
f(x)
0
20
40
60
80
100
x
import mpmath
from mpmath import *

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)

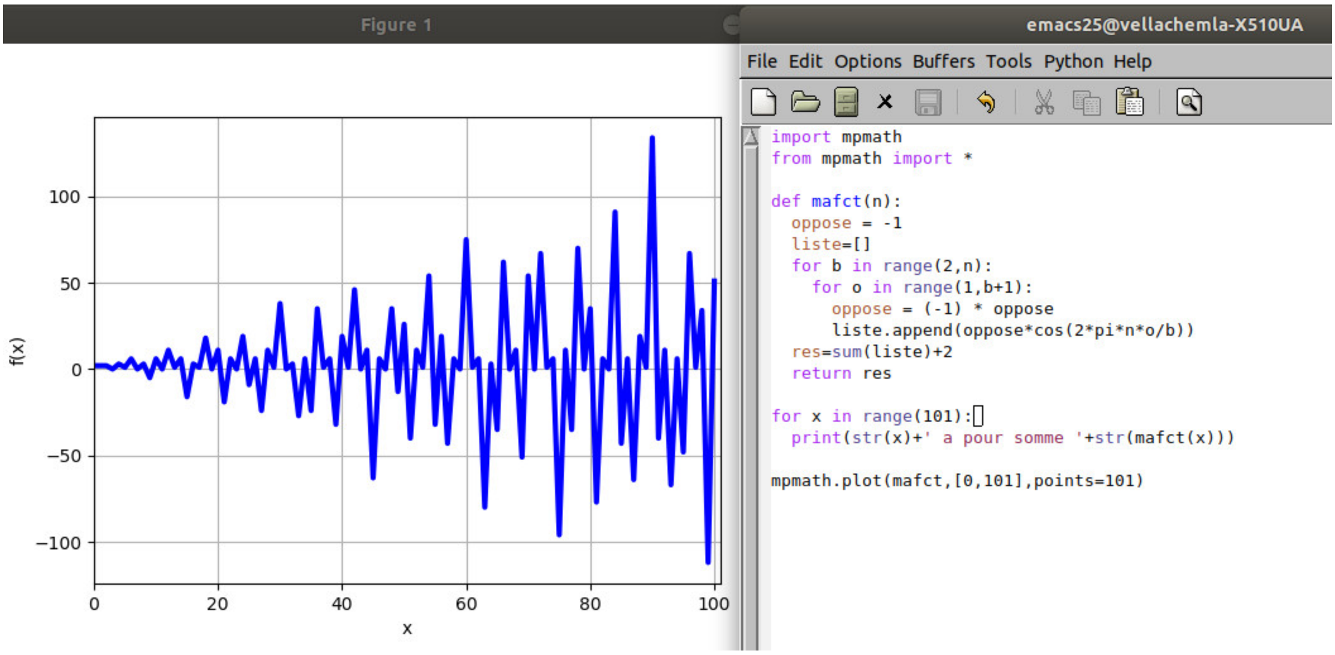


Figure 1
emacs25@vellachemla-X510UA
File Edit Options Buffers Tools Python Help
100
50
0
−50
−100
f(x)
0
20
40
60
80
100
x
import mpmath
from mpmath import *

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 $\frac{1}{2}$ pour les restes modulaires vus comme des fractions rationnelles.

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 $\dfrac{2}{p-1}$ et b est égal à $\dfrac{3p+1}{2p-2}$.

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 :

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

Surprise par une somme alternée de cosinus quotientés (Denise Vella-Chemla, 20.11.2019)

On vient de découvrir les résultats d'un programme qui nous époustouflent. Les voici : en alternant une somme de cosinus, et en ne conservant que les cosinus égaux à 1 ou -1, on obtient une fonction qui associe aux nombres premiers de la forme $n = 4k + 1$ une image égale à $\dfrac{n-1}{2} - 2$ tandis que cette fonction associe aux nombres premiers de la forme $n = 4k + 3$ une image égale à $\dfrac{n-1}{2}$ et qu'enfin, elle associe des nombres différents de ces deux expressions aux nombres composés.

Voici le programme en python qui calcule la fonction en question :

```
import math
from math import atan, cos

PI = 4.0 * atan(1.0)
print int(-1)␣=␣+str(int(-1))
print int(1)␣=␣+str(int(1))
print int(-0.6)␣=␣+str(int(-0.6))
print int(-0.2)␣=␣+str(int(-0.2))
print int(-0.5)␣=␣+str(int(-0.5))
print int(0.6)␣=␣+str(int(0.6))
print int(0.2)␣=␣+str(int(0.2))
print int(0.5)␣=␣+str(int(0.5))

for n in range(2,101):
  oppose = 1
  somme = 0.0
  chaine=␣
  for i in range(2,n):
    sommeinterm = 0.0
    for j in range(1,i+1):
      oppose = (-1)*oppose
      sommeinterm += oppose*int(cos(2.0 * PI * float(n) * float(j) / float(i)))
      somme += oppose*int(cos(2.0 * PI * float(n) * float(j) / float(i)))
      print(str(n)+"␣somme␣globale␣␣␣"+str(somme-1))
```

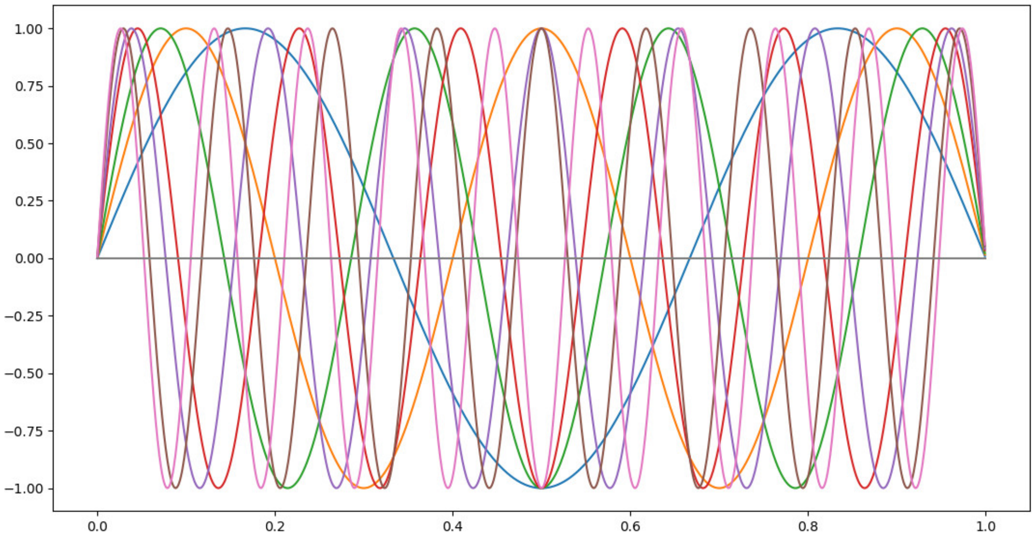
Voici les images des nombres de 1 à 100 fournies par la fonction :

sac(1) = -1	sac(21) = 28	sac(41) = 18	sac(61) = 28	sac(81) = 126
sac(2) = -1	sac(22) = -13	sac(42) = -61	sac(62) = -33	sac(82) = -41
sac(3) = 1	sac(23) = 11	sac(43) = 21	sac(63) = 123	sac(83) = 41
sac(4) = -2	sac(24) = -18	sac(44) = -26	sac(64) = -2	sac(84) = -122
sac(5) = 0	sac(25) = 18	sac(45) = 80	sac(65) = 78	sac(85) = 104
sac(6) = -5	sac(26) = -13	sac(46) = -25	sac(66) = -97	sac(86) = -45
sac(7) = 3	sac(27) = 33	sac(47) = 23	sac(67) = 33	sac(87) = 127
sac(8) = -2	sac(28) = -18	sac(48) = -34	sac(68) = -34	sac(88) = -50
sac(9) = 6	sac(29) = 12	sac(49) = 46	sac(69) = 100	sac(89) = 42
sac(10) = -5	sac(30) = -41	sac(50) = -41	sac(70) = -93	sac(90) = -165
sac(11) = 5	sac(31) = 15	sac(51) = 73	sac(71) = 35	sac(91) = 129
sac(12) = -10	sac(32) = -2	sac(52) = -26	sac(72) = -66	sac(92) = -50
sac(13) = 4	sac(33) = 46	sac(53) = 24	sac(73) = 34	sac(93) = 136
sac(14) = -9	sac(34) = -17	sac(54) = -69	sac(74) = -37	sac(94) = -49
sac(15) = 19	sac(35) = 45	sac(55) = 71	sac(75) = 149	sac(95) = 123
sac(16) = -2	sac(36) = -34	sac(56) = -34	sac(76) = -42	sac(96) = -66
sac(17) = 6	sac(37) = 16	sac(57) = 82	sac(77) = 112	sac(97) = 46
sac(18) = -17	sac(38) = -21	sac(58) = -29	sac(78) = -113	sac(98) = -97
sac(19) = 9	sac(39) = 55	sac(59) = 29	sac(79) = 39	sac(99) = 197
sac(20) = -10	sac(40) = -18	sac(60) = -82	sac(80) = -34	sac(100) = -82

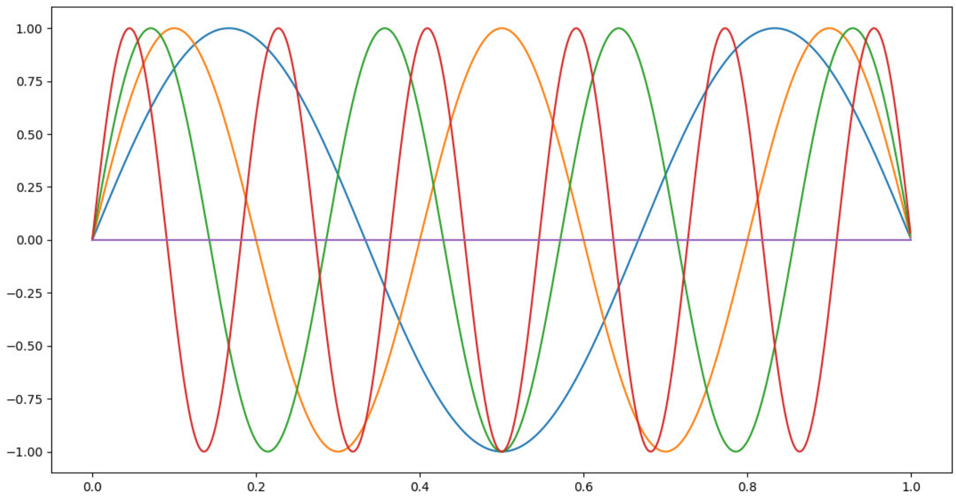
On est étonné de voir que la fonction associe l'opposé $-p$ d'un nombre premier p à son double $2p$ lorsque c'est un nombre premier de la forme $4k+1$ (voir les images $sac(10) = -5, sac(26) = -13, sac(34), sac(58), sac(74),$ $sac(82), \ldots$).

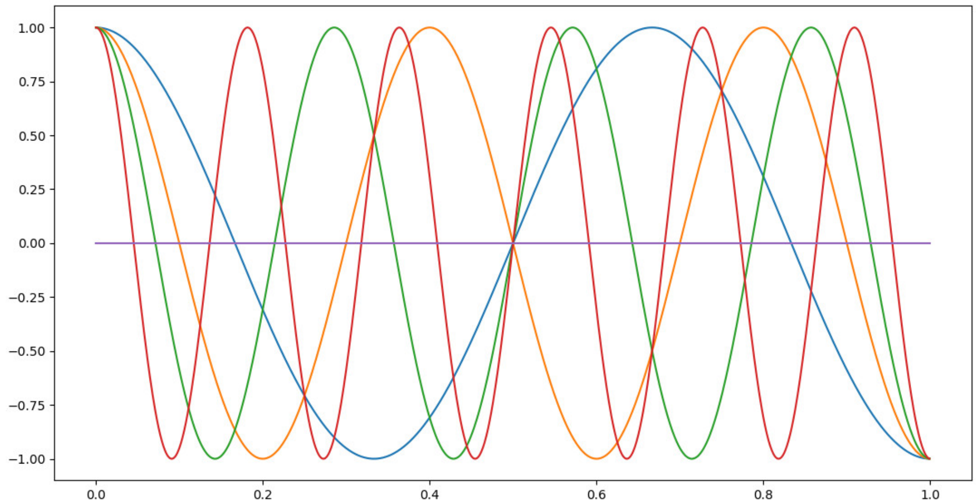
Aux doubles de nombres premiers $2p$ avec p de la forme $4k+3$, la fonction associe $-p-2$ (voir les images $sac(6) = -5, sac(14) = -9, sac(22) = -13, sac(38), sac(46), sac(62), sac(86), sac(94), \ldots$).

Ci-dessous des graphiques montrant vraisemblablement pourquoi les $4k+1$ et les $4k+3$ présentent un comportement différent : dans l'intervalle $[0, 1]$, pour $\dfrac{1}{2}$, les sinusoïdes des nombres premiers de la forme $4k+1$ se croisent "en haut" du graphique tandis que celles des nombres premiers de la forme $4k+3$ se croisent "en bas". Le premier graphique ci-dessous montre des sinusoïdes et non des cosinusoïdes.



Si l'on se cantonne aux nombres premiers 3, 5, 7, et 11 pour gagner en lisibilité, voici les graphiques des sinusoïdes et ceux des cosinusoïdes.





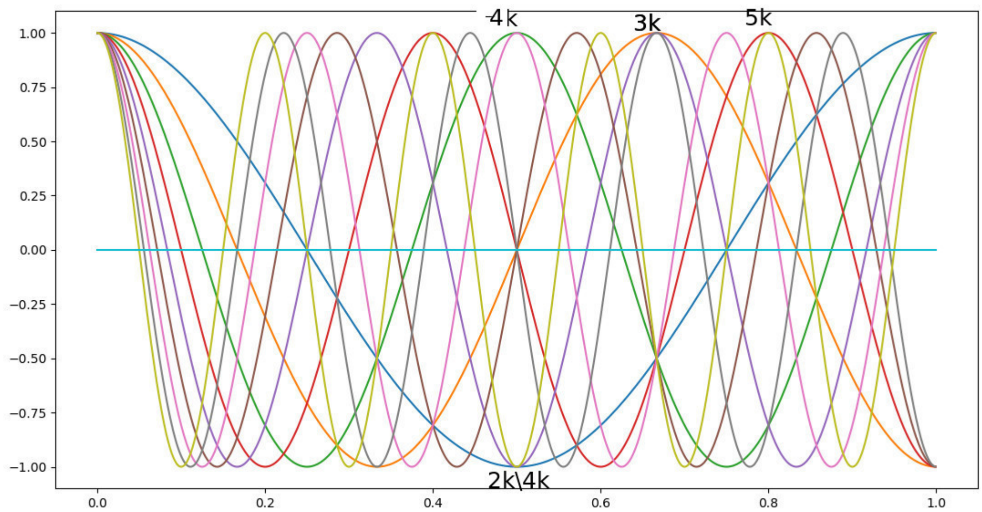
Toutes les cosinusoïdes de fonctions de la forme $\cos(k\pi x)$ avec k impair se croisent en 1/2.

Les nombres premiers se "voient-ils" sur un ensemble de cosinusoïdes ?

Voici le programme en python qui visualise les cosinusoïdes de $cos(2\pi x)$ à $cos(10\pi x)$:

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0.0,1.0,0.001)
x2 = np.cos(2*np.pi*x) ; x3 = np.cos(3*np.pi*x) ;
x4 = np.cos(4*np.pi*x) ; x5 = np.cos(5*np.pi*x) ;
x6 = np.cos(6*np.pi*x) ; x7 = np.cos(7*np.pi*x) ;
x8 = np.cos(8*np.pi*x) ; x9 = np.cos(9*np.pi*x) ;
x10 = np.cos(10*np.pi*x)
tt = [0 for k in x]
plt.plot(x, x2) ; plt.plot(x, x3) ;
plt.plot(x, x4) ; plt.plot(x, x5) ;
plt.plot(x, x6) ; plt.plot(x, x7) ;
plt.plot(x, x8) ; plt.plot(x, x9) ;
plt.plot(x, x10)
plt.plot(x,tt)
plt.show()
```

Dans le graphique résultant, que voit-on ?

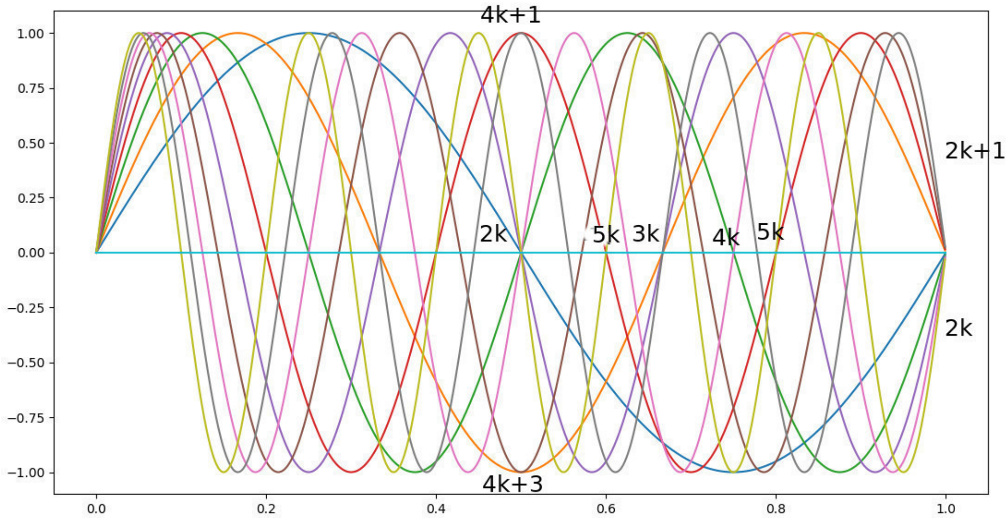


En abscisse 1/2, les courbes des entiers pairs se croisent en haut (ordonnée = +1) ou en bas (ordonnée = -1), les courbes des entiers impairs se croisent sur l'axe des ordonnées (ordonnée = 0). Le point commun à plusieurs courbes en haut au centre voit se croiser les courbes des entiers $4k$ (là, 4 et 8). En bas à la même abscisse de 1/2 on voit les courbes de la forme $2k\backslash 4k$, on désigne par cette notation les multiples de 2 non divisibles par 4 (là, 2, 6 et 10).

En haut, à droite du milieu, en abscisse 2/3, on voit les courbes des $3k$; si on descend verticalement à même abscisse, il n'y a pas de croisement de plusieurs courbes tout en bas et 3 est premier. Idem pour 5 un peu plus loin. Mais pour 7, comme 14 est supérieur à 10, on ne voit pas de croisement en abscisse 6/7.

On peut peut-être utiliser ces points multiples pour compter les nombres premiers ; ici le nombre de nombres premiers impairs compris entre 3 et 5 la moitié de 10 est 2, le nombre de points multiples sur la portion de la droite correspondant à l'ordonnée +1 et pour une abscisse $> 1/2$. Les nombres premiers correspondent aux points multiples d'ordonnée 1 (en haut du diagramme) qui ne tombent pas "en face" de points multiples d'ordonnée -1 (en bas), cette idée permet d'éliminer le nombre 4 (collé à 8) sous prétexte qu'il est "en face" des multiples de son diviseur 2.

Sur une visualisation utilisant plutôt des courbes de sinus, les points multiples sont amenés sur l'axe des ordonnées.



Approche purement syntaxique (Denise Vella-Chemla, 12.1.2017)

1) Modéliser

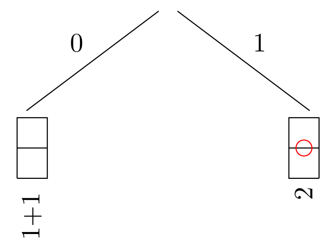
On cherche ce que les nombres premiers symétrisent.

On sait par exemple qu'un nombre premier p a exactement $\frac{p-1}{2}$ résidus quadratiques, tandis que ce n'est pas le cas d'un nombre composé. Si on considère x et $p-x$ inférieurs à p, soit tous 2 sont résidus quadratiques de p simultanément, soit si l'un l'est, l'autre ne l'est pas et inversement. On peut voir dans ces faits une forme de symétrie.

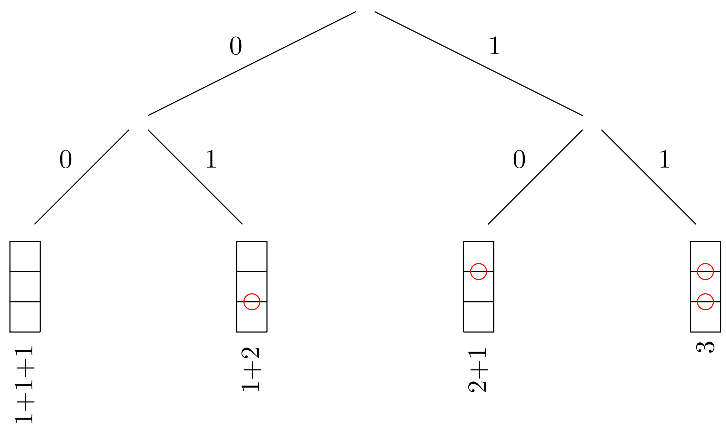
Cherchons d'autres formes de symétrie en étudiant les compositions des nombres. On prend comme référence l'article de wikipedia https://fr.wikipedia.org/wiki/Composition$_($*combinatoire*$)$.

Un nombre n a 2^{n-1} compositions différentes. On peut voir les compositions de n comme feuilles d'un arbre binaire de hauteur $n-1$. On voit que l'ordre des sommants importe, 1+1+2+2 et 1+2+1+2 sont des compositions différentes de 6 (alors qu'elles correspondent à la même partition). Il y a une correspondance bijective entre les mots booléens de $n-1$ caractères et les compositions de n. Par exemple, le mot de 5 lettres 10010 correspond à la décomposition $2+1+2+1$ de 6.

Arbre binaire des compositions de 2



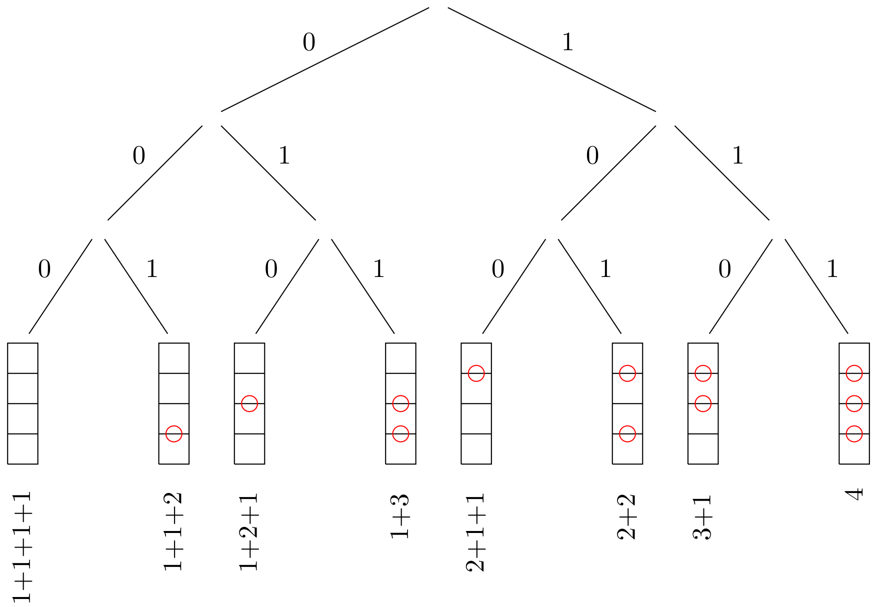
Arbre binaire des compositions de 3 (lire les dessins de haut en bas)



Pour obtenir une composition de $n+1$ à partir d'une composition de n, il suffit soit d'ajouter un sommant 1+ au début de la composition de n, soit de remplacer le premier sommant de la composition de n par son successeur, en obtenant à partir de la composition $n = s_0 + A$ la composition $n+1 = (1+s_0) + A$, le calcul entre parenthèses devant être effectivement effectué.

Voyons cela pour le passage des compositions de 3 à celles de 4 : de l'ensemble $\{1+1+1, 1+2, 2+1, 3\}$ des compositions de 3, on peut concaténer 1+ au début de chaque somme, ce qui permet d'obtenir l'ensemble de compositions de 4 $\{1+1+1+1, 1+1+2, 1+2+1, 1+3\}$; ou bien dans chaque composition de 3, on remplace le premier sommant par son successeur, ce qui permet d'obtenir l'ensemble de compositions de 4 $\{2+1+1, 2+2, 3+1, 4\}$. On réitère le processus pour passer de l'ensemble des compositions de 4 $\{1+1+1+1, 1+1+2, 1+2+1, 1+3, 2+1+1, 2+2, 3+1, 4\}$ à l'ensemble de compositions de 5 constitué de l'union des deux ensembles $\{1+1+1+1+1, 1+1+1+2, 1+1+2+1, 1+1+3, 1+2+1+1, 1+2+2, 1+3+1, 1+4\}$ et $\{2+1+1+1, 2+1+2, 2+2+1, 2+3, 3+1+1, 3+2, 4+1, 5\}$.

Arbre binaire des compositions de 4



On appelle compositions triviales les deux compositions constituées l'une uniquement de n caractères 1 séparés par des signes "+", de la forme 1+1+...+1, et l'autre de n seul. Un nombre composé est notamment caractérisé par le fait que l'une de ses compositions au moins, différente des deux compositions triviales, est telle que quelle que soit la permutation qu'on pourrait effectuer entre 2 de ses sommants, cette composition reste identique à elle-même. Il en est ainsi de la composition 2+2+2 de 6, ou 5+5+5+5+5 de 25. Un nombre premier n'a aucune de ses compositions qui est ainsi invariante par toute permutation de ses sommants. Cette caractérisation de la primalité est exclusivement syntaxique.

2) Essayer de formaliser

A chaque entier est associé un ensemble de parties de $\mathbb{N}$. Chaque partie de $\mathbb{N}$ est une application de $\mathbb{N}$ dans $\{0,1\}$ et on code par un booléen le fait qu'une partie soit image d'un entier ou pas.

Une partie de $\mathbb{N}$ étant codée par un mot infini, on complète chaque mot booléen de la section précédente par une infinité de 0 ; cette infinité de zéros à droite n'ajoute pas de sommant à l'écriture additive.

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

Voyons en quoi consiste le passage d'un entier au suivant selon cette formalisation (informatiquement, c'est trivial, ça consiste à concaténer une lettre 0 ou 1 à gauche des mots de n).

On a 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 $d_n : \mathbb{N} \longrightarrow \{0,1\}$ et

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

La condition correspondant au fait d'être composé pour un nombre entier qui consiste à avoir une composition (différente des deux compositions triviales) de la forme $x+x+\ldots+x+x$ se traduit très simplement sur les mots booléens : elle consiste en l'apparition dans le mot d'une puissance au moins carrée d'un sous-mot non-nul (le sous-mot en question est appelé période en théorie des langages rationnels) ; par exemple, la composition $2+2+2$ de 6, codée par le mot $1010100000\ldots$ contient le sous-mot 10 répété trois fois, elle s'écrit $(10)^3 0^\infty$.

Annexe 1 : les 32 compositions de 6 et leur mot booléen correspondant

$1+1+1+1+1+1$ (00000)	$2+1+1+1+1$ (10000)
$1+1+1+1+2$ (00001)	$2+1+1+2$ (10001)
$1+1+1+2+1$ (00010)	$2+1+2+1$ (10010)
$1+1+1+3$ (00011)	$2+1+3$ (10011)
$1+1+2+1+1$ (00100)	$2+2+1+1$ (10100)
$1+1+2+2$ (00101)	$2+2+2$ (10101)
$1+1+3+1$ (00110)	$2+3+1$ (10110)
$1+1+4$ (00111)	$2+4$ (10111)
$1+2+1+1+1$ (01000)	$3+1+1+1$ (11000)
$1+2+1+2$ (01001)	$3+1+2$ (11001)
$1+2+2+1$ (01010)	$3+2+1$ (11010)
$1+2+3$ (01011)	$3+3$ (11011)
$1+3+1+1$ (01100)	$4+1+1$ (11100)
$1+3+2$ (01101)	$4+2$ (11101)
$1+4+1$ (01110)	$5+1$ (11110)
$1+5$ (01111)	6 (11111)

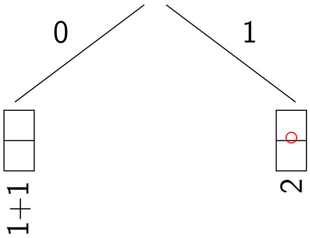
Annexe 2 : Programme de passage des mots booléens aux compositions

```
#include <cstdlib>
#include <iostream>
#include <vector>

using std::cout;
using std::endl;
using std::atoi;

typedef std::vector<bool> bitset;

void print_bits(bitset &bits) {
    for (int i = 0; i < bits.size(); ++i)
        cout << (int) bits[i];
}

void print_deco(bitset &bits) {
    int s = 1;
    for (int i = 0; i < bits.size(); ++i) {
        if (bits[i]) {
            ++s;
        } else {
            cout << s << "+";
            s = 1;
        }
    }
    cout << s;
}
```

```
void generate(int n, bitset &bits) {
    if (n-- > 0) {
        bits[n] = 0; generate(n, bits);
        bits[n] = 1; generate(n, bits);
    } else {
        print_bits(bits);
        cout << " : ";
        print_deco(bits);
        cout << endl;
    }
}

int main(int argc, char* argv[]) {
    int n_max = 0;
    if (argc > 1) n_max = atoi(argv[1]);
    for (int n = 0; n < n_max; ++n) {
        cout << "n = " << n + 1 << " :" << endl;
        bitset bits(n);
        generate(n, bits);
        cout << endl;
    }
}
```

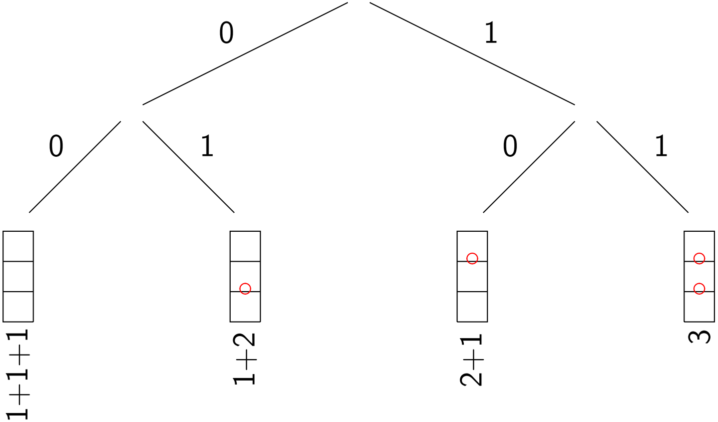
Compositions et mots booléens

- 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 $\mathbb{N}$.

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

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.

Deux par classe (Denise Vella-Chemla, juillet 2022).

1. Caractérisation des décomposants de Goldbach de n supérieurs à $\sqrt{n}$[1]

Soit $n \in 2\mathbb{N}+6$ un entier pair supérieur à 6.

Pour tout $p \in \mathbb{P}^*$ premier impair inférieur à $\sqrt{n}$ (*i.e.* $3 \leq p \leq \sqrt{n}$), on définit l'ensemble :

$$F_n(p) = \{m \in 2\mathbb{N}+1 : 3 \leq m \leq n/2,\ m \not\equiv 0\ [p],\ m \not\equiv n\ [p]\}$$

L'intersection des ensembles $F_n(p)$ pour tout p premier compris entre 3 et $\sqrt{n}$ est notée :

$$D_n = \bigcap_{\substack{p\in\mathbb{P} \\ 3\leq p\leq\sqrt{n}}} F_n(p)$$

Nous allons montrer que D_n et son complémentaire $n-D_n$ ne contiennent que des nombres premiers.
Lemme 1 : Soit $m \in 2\mathbb{N}+1$ un entier impair. Si m n'est divisible par aucun nombre premier compris entre 3 et $\sqrt{m}$, alors il est premier.

Démonstration : Si m est composé, on a $m = pq$, où p est le plus petit nombre premier intervenant dans la factorisation de m en nombres premiers et où q est le produit de tous les autres facteurs. Puisque m est impair, $p \geq 3$, et puisque $q \geq p$ (q étant le produit d'entiers $\geq p$), $m = pq \geq pp = p^2$ et donc $\sqrt{m} \geq p$ (la fonction racine carrée étant croissante). On a ainsi montré que si m impair est composé, il est divisible par un premier compris entre 3 et $\sqrt{m}$. Le lemme s'obtient par contraposition. $\square$

Lemme 2 : $D_n \subseteq \mathbb{P}$

Démonstration : Soit $m \in D_n$. Alors $m \in F_n(p)$ pour tout p premier compris entre 3 et $\sqrt{n}$. Par conséquent, m est impair et m n'est divisible par aucun nombre premier p compris entre 3 et $\sqrt{n}$ (puisque $m \not\equiv 0\ [p]$), et donc *a fortiori* par aucun premier compris entre 3 et $\sqrt{m}$ (car $m \leq n/2 \implies m \leq n \implies \sqrt{m} \leq \sqrt{n}$). D'après le lemme 1, m est donc premier. $\square$

Lemme 3 : $n - D_n \subseteq \mathbb{P}$

Démonstration : Soit $m \in D_n$. Alors $m \in F_n(p)$ pour tout p premier compris entre 3 et $\sqrt{n}$. Par conséquent, $n-m$ est impair (car m est impair et n pair) et $n-m$ n'est divisible par aucun nombre premier p compris entre 3 et $\sqrt{n}$ (puisque $m \not\equiv n\ [p]$), et donc *a fortiori* par aucun premier compris entre 3 et $\sqrt{n-m}$ (car $n-m \leq n \implies \sqrt{n-m} \leq \sqrt{n}$). D'après le lemme 1, $n-m$ est donc premier. $\square$

Les ensembles D_n ne contiennent que des décomposants de Goldbach de n.

[1] Leila Schneps a démontré que la caractérisation de certains décomposants de Goldbach que je proposais était justifiée.

Lemme 4 : Soit $n \in 2\mathbb{N} + 6$. Si $D_n \neq \varnothing$, alors n vérifie la conjecture de Goldbach.

Démonstration : Si $D_n \neq \varnothing$, il contient un entier p nécessairement premier (d'après le lemme 1), tel que $q = n - p$ est également premier (d'après le lemme 2), et donc $n = p + q$ vérifie la conjecture de Goldbach.

2. Minoration du nombre de décomposants de Goldbach

La caractérisation de la Section 1 suggère que $\dfrac{n}{2} \displaystyle\prod_{2<p\leqslant n}\left(1 - \dfrac{2}{p}\right)$ doit minorer le nombre de décomposants de Goldbach d'un nombre pair : les "pires" des cas (ou cas "très criblants") adviennent lorsque le nombre pair est de la forme $2^k p$ avec p un nombre premier, car alors on élimine deux classes de congruences selon tout module premier inférieur à $\sqrt{n}$, et les cas "moins criblants" adviennent lorsque le nombre pair considéré a de nombreux diviseurs (comme le nombre 60 par exemple) car alors on ne doit éliminer qu'une seule classe de congruence au lieu de 2 selon chaque module premier divisant n.

On peut utiliser la formule 2.6 de [1] qui fournit l'estimation :

"(2.6)
$$\begin{aligned}\prod_{\alpha<p\leq x}\left(1-\frac{\alpha}{p}\right) &= \exp\left\{\sum_{\alpha<p\leq x}\log\left(1-\frac{\alpha}{p}\right)\right\}\\ &\cong \exp\left\{c_1(\alpha) - \alpha\sum_{p\leq x}\frac{1}{p}\right\} \cong \frac{c(\alpha)}{(\log\, x)^{\alpha}},\end{aligned}$$

où α est une constante réelle, habituellement prise comme étant égale à 1."

Il est expliqué dans le paragraphe suivant de l'article de Rosser et Schoenfeld qu'il est possible d'utiliser les constantes $c(\alpha)$ et $c_1(\alpha)$ parce que l'erreur absolue tend vers 0 lorsque x tend vers l'infini.

Pour avoir une idée de la constante $c(2)$, il est possible[2] de démontrer que

$$\left(\prod_{2<p\leqslant n} 1 - \frac{2}{p}\right)^{-1} = \frac{1}{4}e^{2\gamma}\Pi_2^{-1}\log^2 n + O(e^{-c\sqrt{\log n}})$$

avec $e = 2.71828, \gamma = 0.5772156649, \Pi_2 = 0.6601618158$[3].

On choisit de minorer le nombre de décomposants de Goldbach de n en utilisant la minoration

$$\begin{aligned}&\#\{3 \leq dg \leq n/2 \mid n = dg + (n - dg) \text{ avec } dg \text{ et } n - dg \text{ premiers}\}\\ &\geq \frac{n}{2}\frac{4\Pi_2}{e^{2\gamma}}\frac{1}{\log^2 n}\end{aligned}$$

[2] Se reporter à https://math.stackexchange.com/questions/22411/computing-the-product-of-p-p-2-over-the-odd-primes?rq=1.

[3] Voir https://mathworld.wolfram.com/TwinPrimesConstant.html).

3. Résultats numériques.

On utilise le programme suivant :

```python
import math

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

for n in range(6,100002,2):
  moitie = int(n/2)
  #if prime(moitie):
  if True:
    print('')
    print(n)
    nbdg = 0
    for p in range(3, moitie+1,2):
      if prime(p) and prime(n-p):
        nbdg += 1
    print('nbdg ', nbdg)
    estimproddespmoinsdeuxsurp = 0.8324290656/(math.log(n)**2)
    print('produit des p moins 2 sur p',estimproddespmoinsdeuxsurp)
    res = n*estimproddespmoinsdeuxsurp/2
    print(' que multiplie n/2 ', res)
    if res < nbdg:
      print('min reussi')
    else:
      print('min rate')
```

FIGURE 1 : Programme de minoration du nombre de décomposants de Goldbach d'un nombre pair.

pour vérifier que la minoration est effective pour tout n un nombre pair compris entre 6 et 10^5.

Le résultat de ce programme est consultable à l'adresse http://denise.vella.chemla.free.fr/respgm-prod-un-moins-deux-sur-p.pdf.

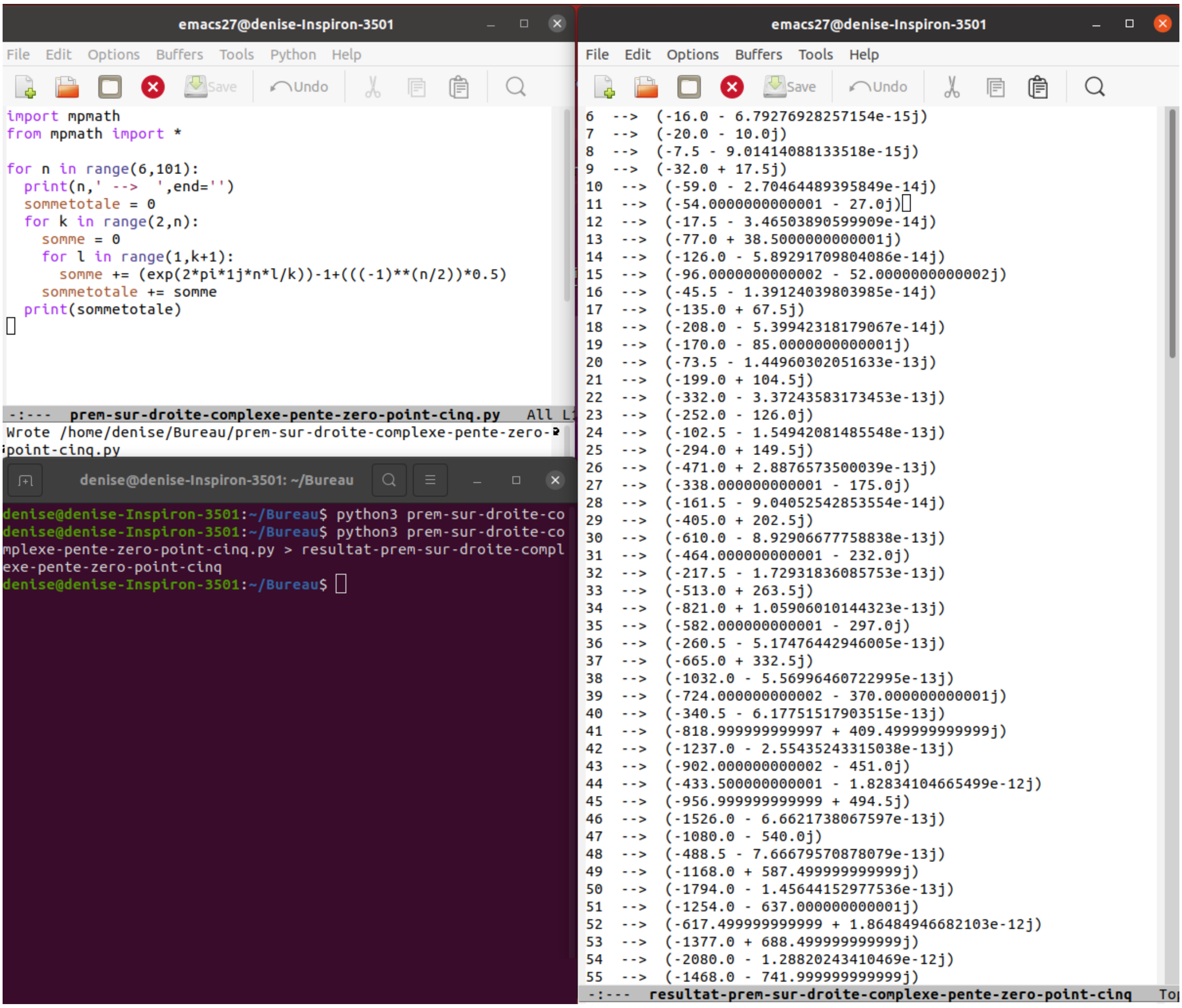
Référence

[1] J. B. Rosser, L. Schoenfeld, *Approximate formulas for some functions of prime numbers*, Illinois J. Math., 6 (1962) 64-94.

Aligner dans le plan complexe les images des nombres premiers par une fonction assez bizarre (Denise Vella-Chemla, 2/10/2022)

On revient sur une fonction définie sur les entiers et qui semble aligner les images des nombres premiers et d'eux seuls sur une droite du plan complexe.

Présentons le programme python calculant cette fonction sur la copie d'écran ci-dessous. Les nombres premiers ont pour images par cette fonction des nombres complexes de la forme $a + \frac{1}{2} a\, i$ tandis que les images des nombres composés semblent être systématiquement extérieures à cette droite.



On avait étudié cette fonction après des travaux autour de la somme des diviseurs, suite à la lecture d'un article d'Euler *Découverte d'une loi tout extraordinaire des nombres par rapport à la somme de leurs diviseurs.*

On peut calculer la somme des diviseurs des nombres comme une somme de cosinus. Pour $n \geq 1$, on définit :

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

Pour $1 \leq k \leq n$, on a :

$$\sigma_k(n) = \begin{cases} \displaystyle\sum_{l=1}^{k} e^{\frac{2i\pi nl}{k}} = k & \text{si } \ k \mid n \\ \displaystyle\sum_{l=1}^{k} e^{\frac{2i\pi nl}{k}} = \sum_{l=1}^{k} \left(e^{\frac{2i\pi n}{k}}\right)^l = \frac{\left(e^{\frac{2i\pi n}{k}}\right)^{k+1} - 1}{e^{\frac{2i\pi n}{k}} - 1} = e^{\frac{2i\pi n}{k}} \left(\frac{e^{2i\pi n} - 1}{e^{\frac{2i\pi n}{k}} - 1}\right) = 0 & \text{sinon} \end{cases}$$

On en déduit que $\sigma(n)$ est la somme des diviseurs de n :

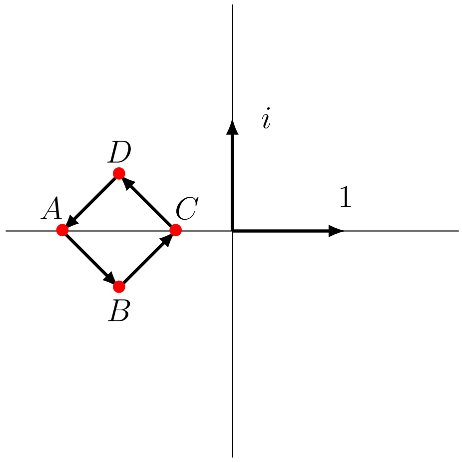
$$\sigma(n) = \sum_{k=1}^{n} \sigma_k(n) = \sum_{k|n} k$$

Fin 2018, on avait eu comme idée[1] de calculer une somme alternée des mêmes cosinus, mais en voulant simplifier cette formule et en commettant plusieurs erreurs, on aboutit à la définition suivante d'une fonction $f_D(n)$ définie sur les entiers (pour $n \geq 1$) :

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

L'ajout du terme bizarre $\left[-1 + \frac{1}{2}\left((-1)^{\frac{n}{2}}\right)\right]$ en fin de formule consiste à ajouter à la somme des diviseurs l'un des 4 complexes qui sont les affixes des sommets du carré $(A = -1.5,\ B = -1 - 0.5i,\ C = -0.5,\ D = -1 + 0.5i)$ (*remarque* : on a soustrait de la somme des diviseurs les deux diviseurs triviaux que sont 1 et n : pour effectuer cette soustraction la somme englobante d'indice k est prise pour k variant de 2 à $n - 1$).

[1] Voir http://denise.vella.chemla.free.fr/premiers-image-un-demi.pdf.



On ne sait pas pour l'instant pourquoi la fonction $f_D(n)$ envoie les nombres premiers sur la droite complexe $a + \frac{1}{2}a\,i$.

Condenser (Denise Vella-Chemla, 8.10.2022)[1]

On fournit le code d'un programme court qui distingue les nombres premiers des nombres composés par leur image par une fonction simple :

$$f_D(n) = \sum_{k=2}^{n-1} \sum_{l=1}^{k} \exp\left(\frac{2\pi\ inl}{k}\right) + \left(\frac{1}{2} i^n - 1\right)$$

Pour n impair premier, $\mathfrak{Re}(f_D(n)) = 2 \times \mathfrak{Im}(f_D(n))$, ce qui n'est pas le cas pour n impair composé.

On avait étudié une fonction similaire après des travaux autour de la somme des diviseurs suite à la lecture d'un article d'Euler *Découverte d'une loi tout extraordinaire des nombres par rapport à la somme de leurs diviseurs.*

On peut calculer la somme des diviseurs des nombres comme une somme de cosinus.

Pour $n \geq 1$, on définit :

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

Pour $1 \leq k \leq n$, on a :

$$\sigma_k(n) = \begin{cases} \displaystyle\sum_{l=1}^{k} \exp\left(\frac{2i\pi nl}{k}\right) & = k \ \text{ si } \ k|n; \\ \displaystyle\sum_{l=1}^{k} \exp\left(\frac{2i\pi nl}{k}\right) = \sum_{l=1}^{k} \left(\exp\left(\frac{2i\pi n}{k}\right)\right)^l & \\ \qquad = \dfrac{\left(\exp\left(\frac{2i\pi n}{k}\right)\right)^{k+1} - 1}{\exp\left(\frac{2i\pi n}{k}\right) - 1} & \\ \qquad = \exp\left(\frac{2i\pi n}{k}\right) \left(\dfrac{\exp(2i\pi n) - 1}{\exp\left(\frac{2i\pi n}{k}\right) - 1}\right) & = 0 \ \text{ sinon} \end{cases}$$

On en déduit que $\sigma(n)$ est la somme des diviseurs de n :

$$\sigma(n) = \sum_{k=1}^{n} \mathfrak{Re}(\sigma_k(n)) = \sum_{k|n} k.$$

En ajoutant $\frac{1}{2} i^k - 1$ à chacun des termes de $\sigma(n)$, et en ne considérant que les diviseurs non triviaux de

[1] Merci Jacques.

n, on a :

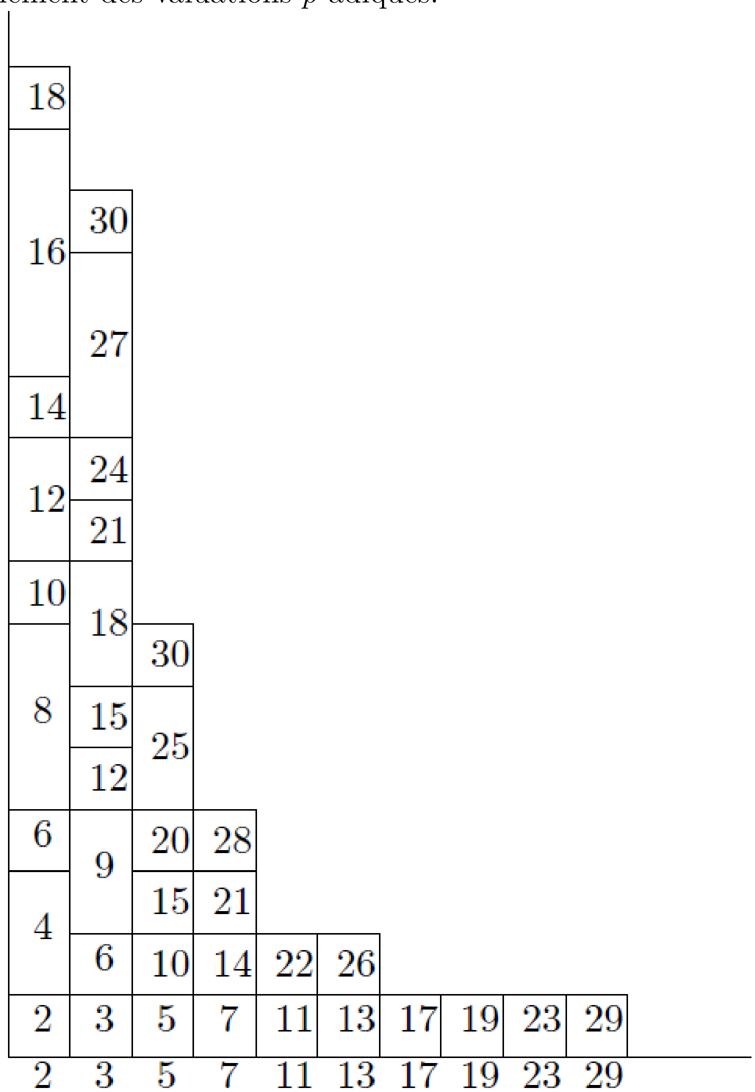
$$\begin{aligned}
\sum_{k=2}^{n-1}\sum_{l=1}^{k}\left(e^{\frac{2i\pi nl}{k}+\frac{1}{2}i^n-1}\right) &= \sum_{k=2}^{n-1}\sum_{l=1}^{k} e^{\frac{2i\pi nl}{k}+\sum_{k=2}^{n-1}\sum_{l=1}^{k}\left(\frac{1}{2}i^n-1\right)} \\
&= \sum_{k=2}^{n-1}\sigma_k(n) + \sum_{k=2}^{n-1} k(\tfrac{1}{2}i^n - 1) \\
&= \sum_{k=2}^{n-1}\sigma_k(n) + (\tfrac{1}{2}i^n - 1)\sum_{k=2}^{n-1} k \\
&= \sigma(n) - \sigma_1(n) - \sigma_n(n) + (\tfrac{1}{2}i^n - 1)(\tfrac{1}{2}n(n-1) - 1) \\
&= \sigma(n) - 1 - n + (\tfrac{1}{2}i^n - 1)(\tfrac{1}{2}n^2 - \tfrac{1}{2}n - 1) \\
&= \sigma(n) - \tfrac{1}{2}n^2 - \tfrac{1}{2}n + \tfrac{1}{2}i^n(\tfrac{1}{2}n^2 - \tfrac{1}{2}n - 1) \\
&= \begin{cases}
\sigma(n) - \frac{1}{4}n^2 - \frac{3}{4}n - \frac{1}{2} & \text{si} \quad n \mod 4 = 0 \\
\sigma(n) - \frac{1}{2}n^2 - \frac{1}{2}n + \frac{1}{2}i(\frac{1}{2}n^2 - \frac{1}{2}n - 1) & \text{si} \quad n \mod 4 = 1 \\
\sigma(n) - \frac{3}{4}n^2 - \frac{1}{4}n + \frac{1}{2} & \text{si} \quad n \mod 4 = 2 \\
\sigma(n) - \frac{1}{2}n^2 - \frac{1}{2}n - \frac{1}{2}i(\frac{1}{2}n^2 - \frac{1}{2}n - 1) & \text{si} \quad n \mod 4 = 3
\end{cases}
\end{aligned}$$

La distinction entre premiers et composés ne provient donc que de la seule somme des diviseurs, égale à $n+1$ pour n premier et strictement supérieure à $n+1$ sinon.

Fonctions en lien avec les nombres premiers

Denise Vella-Chemla

L'empilement des valuations p-adiques.



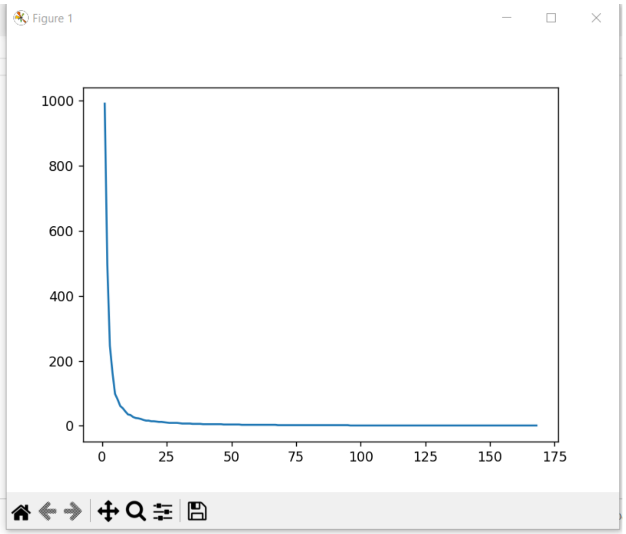
Courbe hyperbolique d'équation $xy = \cancel{nlog}(n)$

L'empilement des valuations p-adiques en python.

rapidement-decroissante.py

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > rapidement-decroissante.py > ...

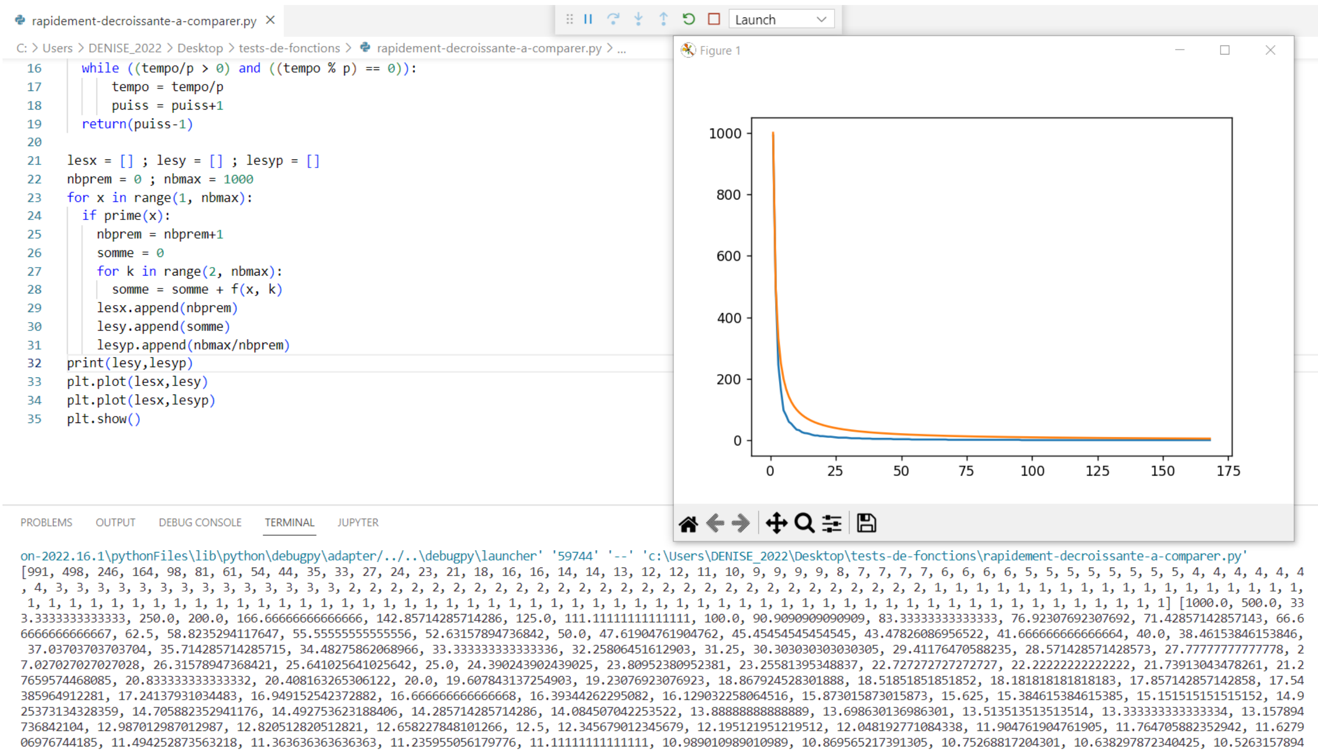
```python
import matplotlib.pyplot as plt

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

def f(p,n):
  puiss = 1
  tempo = n ;
  while ((tempo/p > 0) and ((tempo % p) == 0)):
      tempo = tempo/p
      puiss = puiss+1
  return(puiss-1)

lesx = [] ; lesy = []
nbprem = 0 ; nbmax = 1000
for x in range(1, nbmax):
  if prime(x):
```



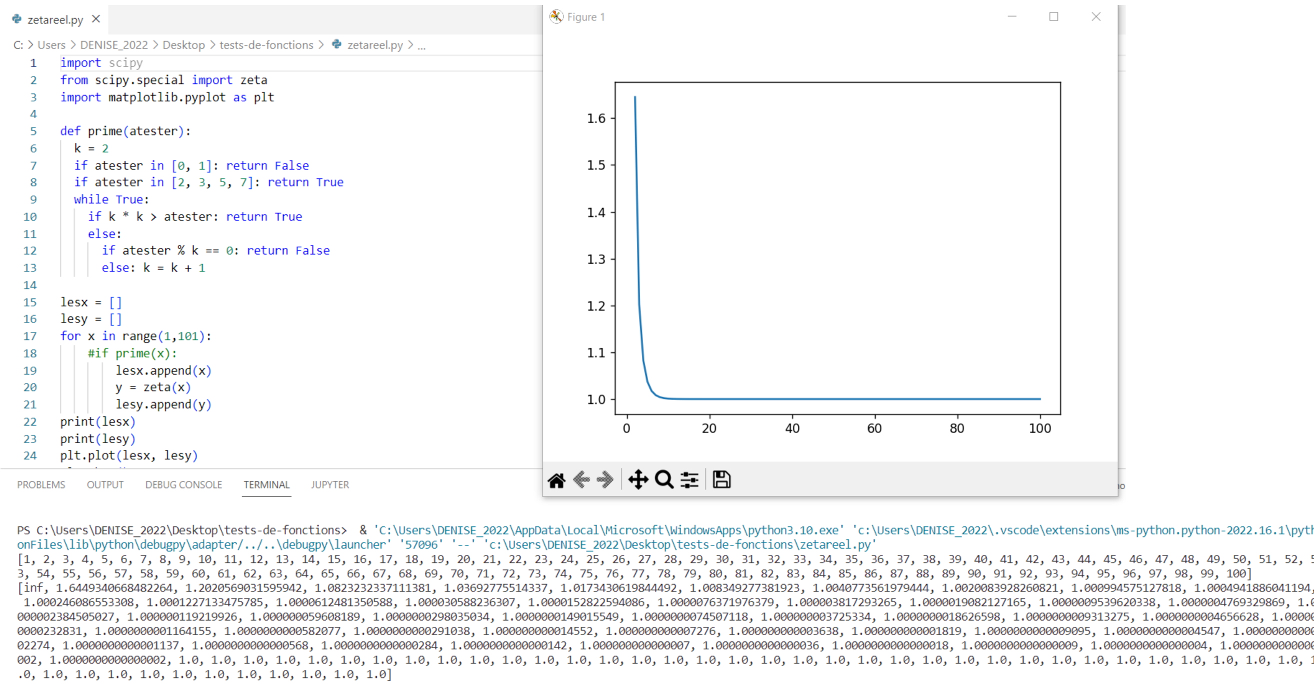
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\DENISE_2022\Desktop\tests-de-fonctions>  & 'C:\Users\DENISE_2022\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\DENISE_2022\.vscode\extensions\ms-python.python-2
022.16.1\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '57136' '--' 'c:\Users\DENISE_2022\Desktop\tests-de-fonctions\rapidement-decroissante.py'
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49,
50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96,
97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 1
35, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168] [991, 498, 246, 164
, 98, 81, 61, 54, 44, 35, 33, 27, 24, 23, 21, 18, 16, 16, 14, 14, 13, 12, 12, 11, 10, 9, 9, 9, 9, 8, 7, 7, 7, 7, 6, 6, 6, 6, 5, 5, 5, 5, 5, 5, 5, 5, 4, 4, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3,
3, 3, 3, 3, 3, 3, 3, 3, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
```

Comparaison empilement des valuations p-adiques et somme des $\frac{nbmax}{p}$ pour les p qui sont les nombres premiers successifs.



Forme de la fonction zeta calculée pour les nombres entiers (images reliées entre elles).

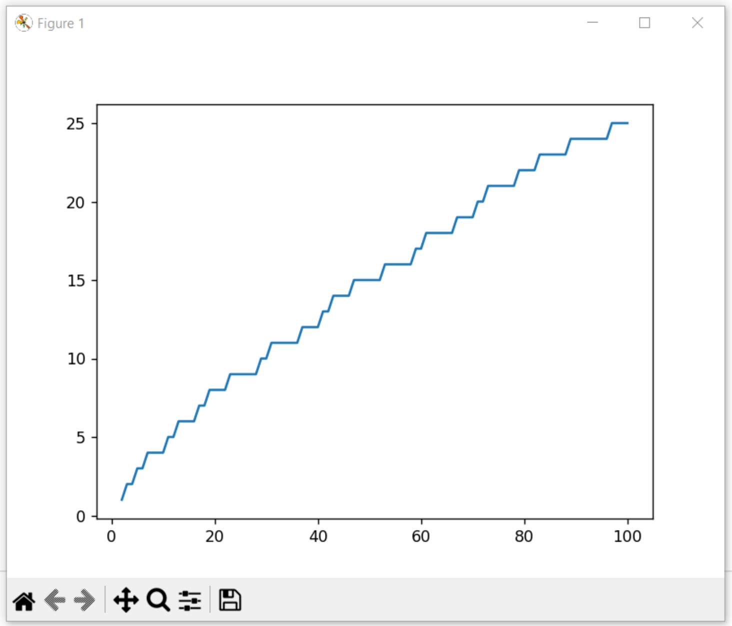


Cumul des inverses des sommes des valuations p-adiques (c'est le graphe de la fonction $\pi(x)$, nombre de nombres premiers inférieurs à x).

cumul-inverses-fct-rapidement-decroissante.py

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > cumul-inverses-fct-rapidement-decroissante.py > ...

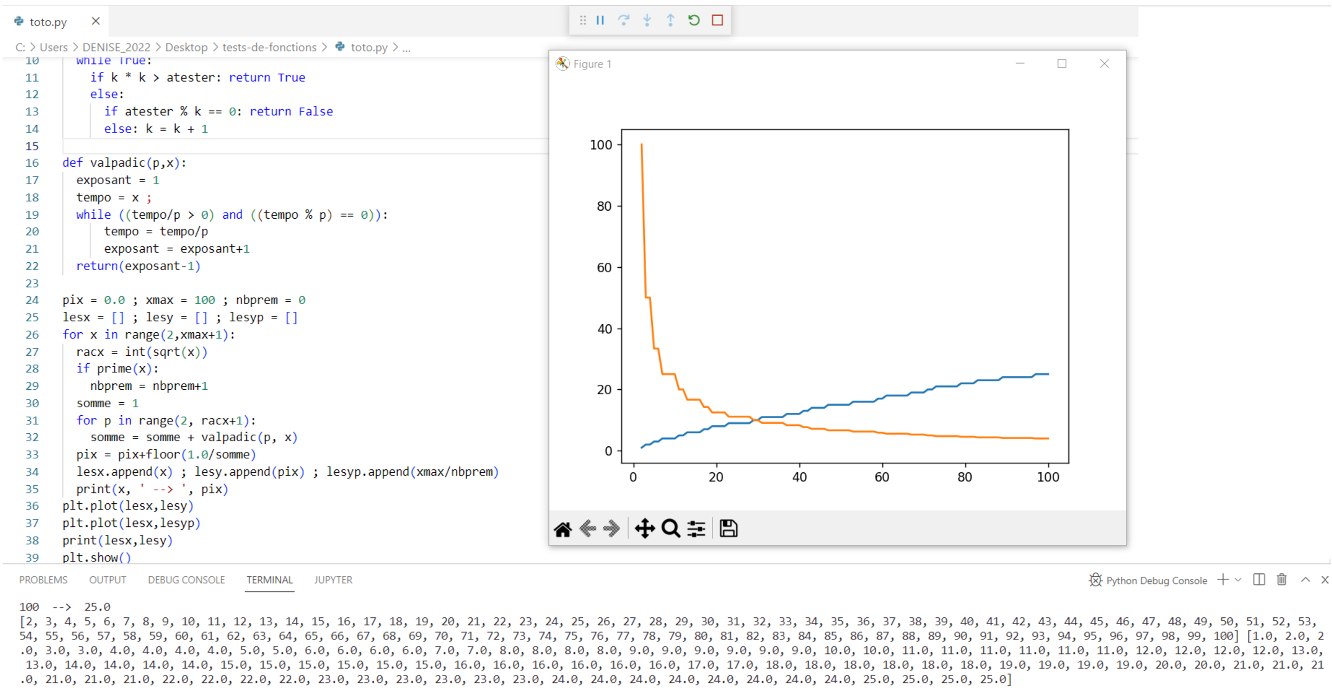
```python
import matplotlib.pyplot as plt
import math
from math import sqrt,floor
import numpy as np

def valpadic(p,x):
  exposant = 1
  tempo = x ;
  while ((tempo/p > 0) and ((tempo % p) == 0)):
    tempo = tempo/p
    exposant = exposant+1
  return(exposant-1)

pix = 0.0 ; xmax = 100
lesx = [] ; lesy = []
for x in range(2,xmax+1):
  racx = int(sqrt(x))
  somme = 1
  for p in range(2, racx+1):
    somme = somme + valpadic(p, x)
  pix = pix+floor(1.0/somme)
  lesx.append(x) ; lesy.append(pix)
  print(x, ' --> ', pix)
plt.plot(lesx,lesy)
plt.show()
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
95  -->  24.0
96  -->  24.0
97  -->  25.0
98  -->  25.0
99  -->  25.0
100  -->  25.0
```

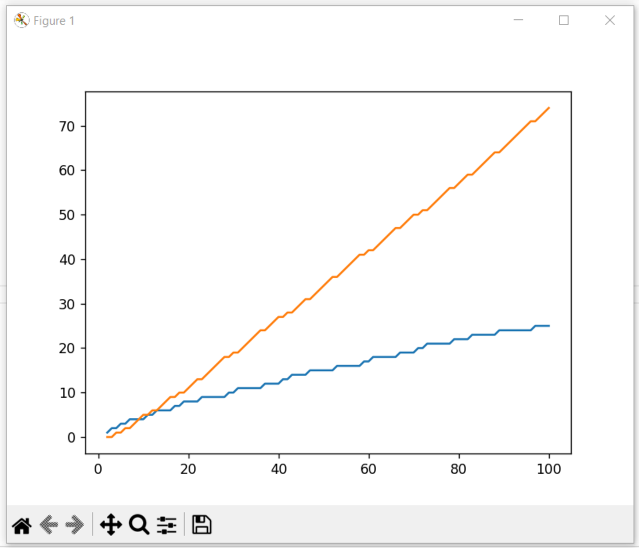
La fonction bleue est $\pi(x)$.
La fonction orange est la courbe d'équation $y = \frac{xmax}{\pi(x)}$.



La fonction bleue est $\pi(x)$.
La fonction orange est la courbe d'équation $\sum \left\lfloor \frac{1}{somme} \right\rfloor$ avec $somme = \sum \frac{nmax}{\pi(x)}$.

toto.py
C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > toto.py > ...

```python
    exposant = 1
    tempo = x ;
    while ((tempo/p > 0) and ((tempo % p) == 0)):
        tempo = tempo/p
        exposant = exposant+1
    return(exposant-1)

pix = 0.0 ; pix2 = 0.0 ; xmax = 100 ; nbprem = 0
lesx = [] ; lesy = [] ; lesyp = []
for x in range(2,xmax+1):
    racx = int(sqrt(x))
    somme = 1
    somme2 = 1
    if prime(x):
      nbprem = nbprem+1
      somme2 = somme2 + (xmax/nbprem)
    for p in range(2, racx+1):
      somme = somme + valpadic(p, x)
    pix = pix + floor(1.0/somme)
    pix2 = pix2 + floor(1.0/somme2)
    lesx.append(x) ; lesy.append(pix) ; lesyp.append(pix2)
    print(x, ' --> ', pix)
plt.plot(lesx,lesy)
plt.plot(lesx,lesyp)
print(lesx,lesy,lesyp)
plt.show()
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

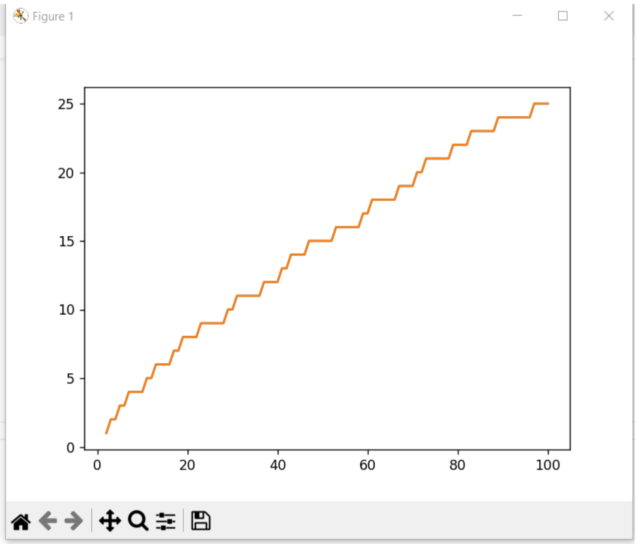
```
.0, 11.0, 11.0, 11.0, 12.0, 12.0, 12.0, 12.0, 13.0, 13.0, 14.0, 14.0, 14.0, 14.0, 15.0, 15.0, 15.0, 15.0, 15.0, 15.0, 16.0, 16.0, 16.0, 16.0, 16.0, 16.0, 17.0, 17.0, 18.0, 18.0, 18.0,
18.0, 18.0, 18.0, 19.0, 19.0, 19.0, 19.0, 20.0, 20.0, 21.0, 21.0, 21.0, 21.0, 21.0, 21.0, 22.0, 22.0, 22.0, 22.0, 23.0, 23.0, 23.0, 23.0, 23.0, 23.0, 24.0, 24.0, 24.0, 24.0, 24.0, 24.0
, 24.0, 24.0, 25.0, 25.0, 25.0, 25.0] [0.0, 0.0, 1.0, 1.0, 2.0, 2.0, 3.0, 4.0, 5.0, 5.0, 6.0, 6.0, 7.0, 8.0, 9.0, 9.0, 10.0, 10.0, 11.0, 12.0, 13.0, 13.0, 14.0, 15.0, 16.0, 17.0, 18.0,
 18.0, 19.0, 19.0, 20.0, 21.0, 22.0, 23.0, 24.0, 24.0, 25.0, 26.0, 27.0, 27.0, 28.0, 28.0, 29.0, 30.0, 31.0, 31.0, 32.0, 33.0, 34.0, 35.0, 36.0, 36.0, 37.0, 38.0, 39.0, 40.0, 41.0, 41.
0, 42.0, 42.0, 43.0, 44.0, 45.0, 46.0, 47.0, 47.0, 48.0, 49.0, 50.0, 50.0, 51.0, 51.0, 52.0, 53.0, 54.0, 55.0, 56.0, 56.0, 57.0, 58.0, 59.0, 59.0, 60.0, 61.0, 62.0, 63.0, 64.0, 64.0, 6
5.0, 66.0, 67.0, 68.0, 69.0, 70.0, 71.0, 71.0, 72.0, 73.0, 74.0]
```

La fonction bleue est $\pi(x)$.
La fonction orange est la courbe qui cumule les $\pi(x)$ avec les $\sum 1 - \frac{xmax}{nbprem}$.

toto.py

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > toto.py > ...

```python
def valpadic(p,x):
  exposant = 1
  tempo = x ;
  while ((tempo/p > 0) and ((tempo % p) == 0)):
      tempo = tempo/p
      exposant = exposant+1
  return(exposant-1)

pix = 0.0 ; pix2 = 0.0 ; xmax = 100 ; nbprem = 0
lesx = [] ; lesy = [] ; lesyp = []
for x in range(2,xmax+1):
  racx = int(sqrt(x))
  somme = 1
  somme2 = 1
  if prime(x):
    nbprem = nbprem+1
    somme2 = somme2 + (xmax/nbprem)
  for p in range(2, racx+1):
    somme = somme + valpadic(p, x)
  pix = pix + floor(1.0/somme)
  pix2 = pix2 + (1.0-floor(1.0/somme2))
  lesx.append(x) ; lesy.append(pix) ; lesyp.append(pix2)
  print(x, ' --> ', pix)
plt.plot(lesx,lesy)
plt.plot(lesx,lesyp)
print(lesx,lesy,lesyp)
plt.show()
```

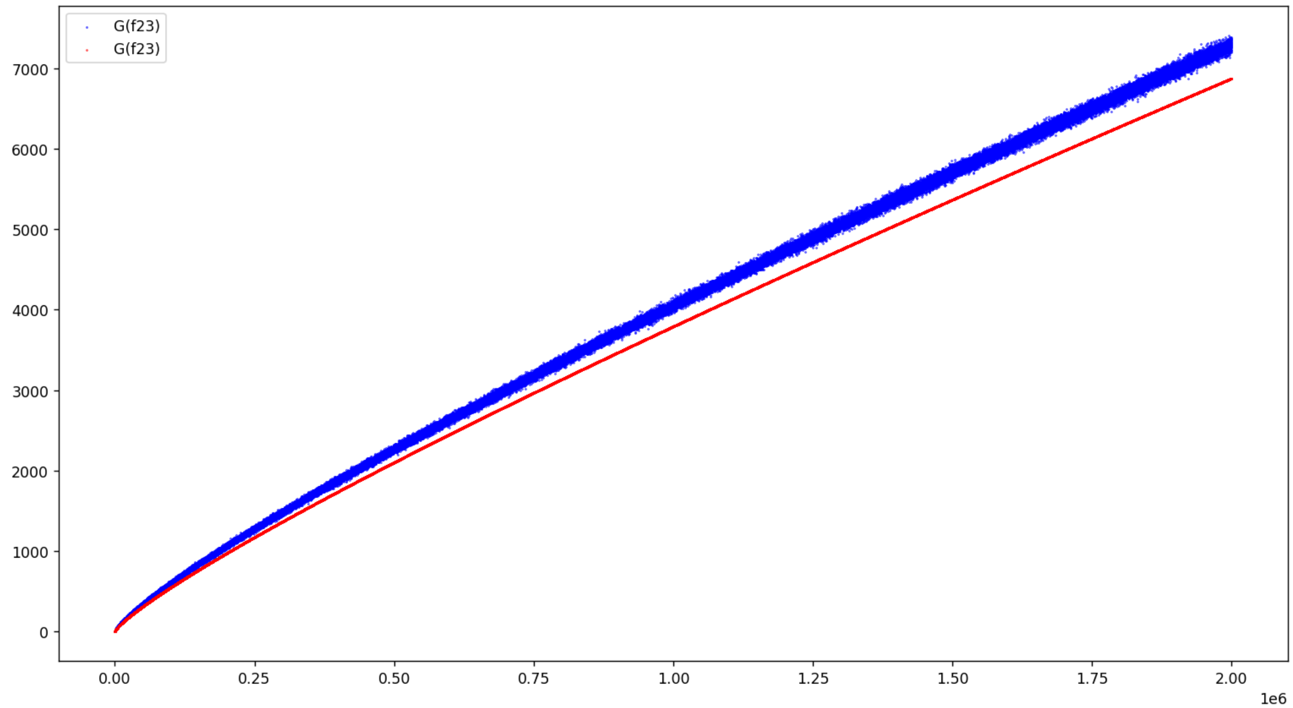


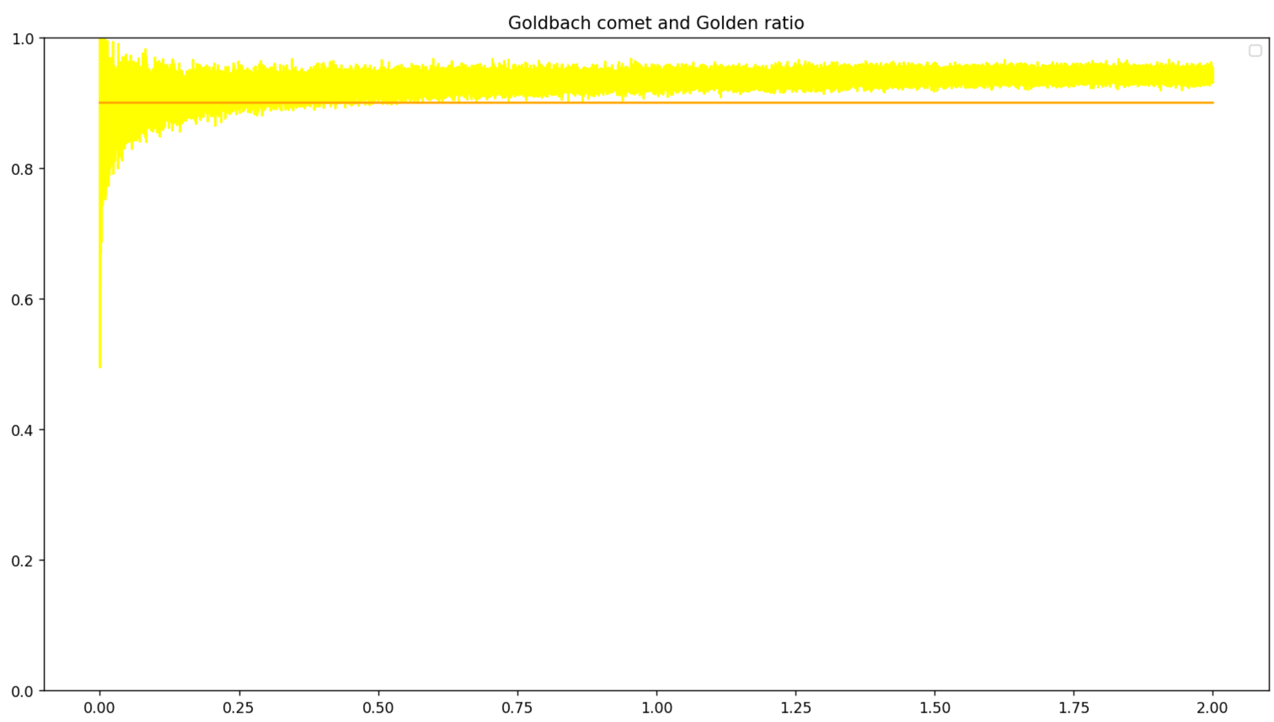
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
.0, 11.0, 11.0, 11.0, 12.0, 12.0, 12.0, 12.0, 13.0, 13.0, 14.0, 14.0, 14.0, 14.0, 15.0, 15.0, 15.0, 15.0, 15.0, 15.0, 16.0, 16.0, 16.0, 16.0, 16.0, 16.0, 17.0, 17.0, 18.0, 18.0, 18.0,
18.0, 18.0, 18.0, 19.0, 19.0, 19.0, 19.0, 20.0, 20.0, 21.0, 21.0, 21.0, 21.0, 21.0, 21.0, 22.0, 22.0, 22.0, 22.0, 23.0, 23.0, 23.0, 23.0, 23.0, 23.0, 24.0, 24.0, 24.0, 24.0, 24.0, 24.0
, 24.0, 24.0, 25.0, 25.0, 25.0, 25.0] [1.0, 2.0, 2.0, 3.0, 3.0, 4.0, 4.0, 4.0, 4.0, 5.0, 5.0, 6.0, 6.0, 6.0, 6.0, 7.0, 7.0, 8.0, 8.0, 8.0, 8.0, 9.0, 9.0, 9.0, 9.0, 9.0, 9.0, 10.0, 10.0
, 11.0, 11.0, 11.0, 11.0, 11.0, 11.0, 12.0, 12.0, 12.0, 12.0, 13.0, 13.0, 14.0, 14.0, 14.0, 14.0, 15.0, 15.0, 15.0, 15.0, 15.0, 15.0, 16.0, 16.0, 16.0, 16.0, 16.0, 16.0, 17.0, 17.0, 18
.0, 18.0, 18.0, 18.0, 18.0, 18.0, 19.0, 19.0, 19.0, 19.0, 20.0, 20.0, 21.0, 21.0, 21.0, 21.0, 21.0, 21.0, 22.0, 22.0, 22.0, 22.0, 23.0, 23.0, 23.0, 23.0, 23.0, 23.0, 24.0, 24.0, 24.0,
24.0, 24.0, 24.0, 24.0, 24.0, 25.0, 25.0, 25.0, 25.0]
```

Comète de Goldbach et Golden ratio (nombre d'or), Denise Vella-Chemla, 25.12.2022

On présente ici deux graphiques :





- le premier présente pour les seuls nombres pairs inférieurs à 2.10^6 de la forme $2p$ avec p un nombre premier (ces nombres pairs vérifient trivialement la conjecture de Goldbach car

$2p = p + p$) leur nombre de décomposants de Goldbach (qu'on notera $G(2p)$) en bleu ; on imagine, sans l'avoir démontré, que les $2p$ (p premier) ont leur nombre de décompositions de Goldbach "en bas de" (i.e. qui minore localement) ce qu'on appelle communément la comète de Goldbach.

En rouge sur ce premier graphique, on a représenté la fonction suivante :

$$f : x \mapsto \frac{(1+\sqrt{5})x}{2\sqrt{5}.\ln\, x.\ln\, x}$$

On constate sur ce premier graphique que la fonction en rouge semble minorer le bas de la comète ;

- le second graphique présente en jaune le ratio $\frac{f(x)}{G(x)}$; on a noté en rouge la droite $y = 0.9$ pour référence. Le ratio semble petit à petit passer au-dessus de la droite de référence.

Si l'on démontrait que les pairs de la forme $2p$ minorent localement la comète, puis que la fonction f minore le nombre de décompositions de Goldbach pour tout pair de la forme $2p$, on aurait une démonstration de la conjecture de Goldbach.

Ce qui serait excitant dans une telle démonstration, ce serait d'utiliser le Golden ratio pour démontrer la conjecture de Goldbach, que d'or, que d'or...

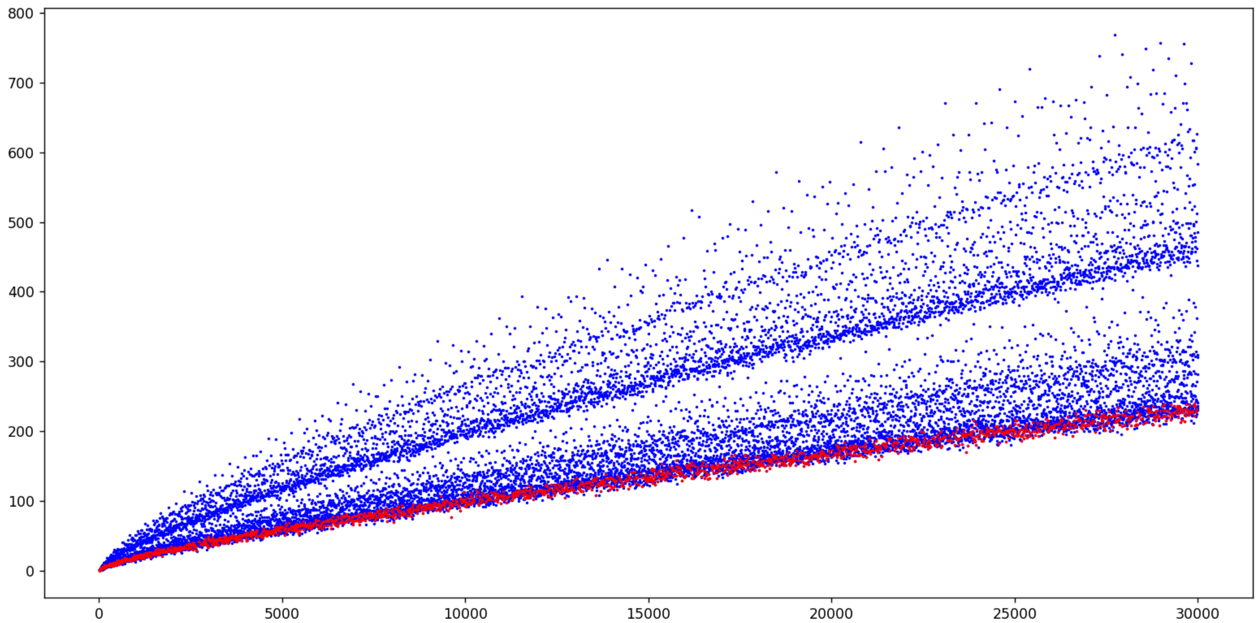
Et le plus extraordinaire encore serait de relier cette idée à un article de Moczurad, Tyszkiewicz et Zaionc [1][1] dans lequel la constante qui intervient ici $\frac{1+\sqrt{5}}{2\sqrt{5}} = 0.7236...$ apparaît comme une limite.

Copions ici un extrait du résumé de leur article :

> "*On considère les types et le lambda-calcul typé sur un nombre fini de types de base. On va rechercher la valeur du ratio du nombre des types tautologiques d'une longueur donnée n rapporté au nombre de tous les types de longueur n. L'objectif de cet article est de trouver la limite de ce ratio lorsque $n \to \infty$. La réponse à cette question équivaut à trouver la "densité" des types tautologiques dans l'ensemble de tous les types, où la probabilité asymptotique, comme on l'appelle, de trouver un type tautologique dans l'ensemble de tous les types. Selon l'isomorphisme de Curry-Howard, cela signifie trouver la densité ou bien la probabilité asymptotique des formules propositionnelles intuitionnistes prouvables parmi l'ensemble de toutes les formules. Pour les types avec un type de base (les formules avec une seule variable propositionnelle), on démontre que la limite existe et qu'elle est égale à $1/2+\sqrt{5}/10$, qui est approximativement égal à 72%. Cela signifie qu'un type aléatoire (en terme de formule) d'une grande taille a environ 72% de chance d'être une tautologie.*".

[1] Voir traduction ici : http://denise.vella.chemla.free.fr/trad-Prop-stat-tautologies.pdf.

Ci-dessous, voyons comme les nombres pairs doubles de nombres premiers ont leur nombre de décompositions de Goldbach qui semblent “minorer localement” la comète de Goldbach :



Références

[1] Malgorzata Moczurad, Jerzy Tyszkiewicz, Marek Zaionc, *Statistical properties of simple types.* Math. Struct. Comput. Sci. 10(5): 575-594 (2000).

Conjecture de Goldbach et logique propositionnelle (propositions à une variable)

Denise Vella-Chemla
janvier 2023

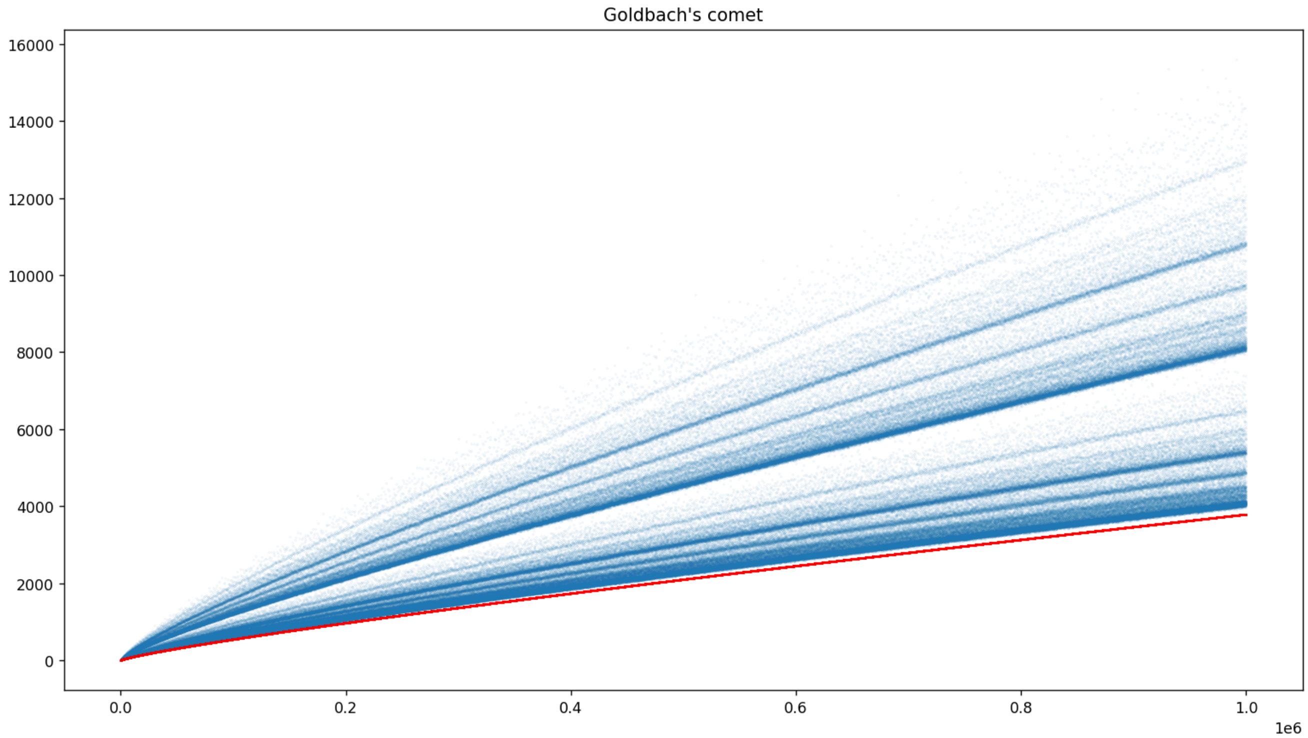
On se situe dans la logique propositionnelle des propositions à une variable. On a des énoncés vrais et des énoncés faux. Si la variable z représente l'énoncé '3+9=10', elle prend la valeur de vérité $faux$. Si la variable z représente l'énoncé '3+35=38', elle prend la valeur de vérité $vrai$. On va dans la suite ne s'intéresser qu'aux énoncés qui sont vrais, de la forme $n = p + q$ avec n un nombre pair, et p et q deux nombres impairs.

Parmi ces énoncés, un certain nombre font intervenir deux nombres premiers en place de p et q. Représentons ces énoncés sur la Figure 1. On a sur la figure ordonné les énoncés dans le plan pour fixer les idées, mais on peut tout à fait les imaginer "en vrac".

6= $3+3$	6= $5+1$	6= $7+(-1)$	6= $9+(-3)$	6= $11+(-5)$	6= $13+(-7)$	6= $15+(-9)$	6= $17+(-11)$
8= $3+5$	8= $5+3$	8= $7+1$	8= $9+(-1)$	8= $11+(-3)$	8= $13+(-5)$	8= $15+(-7)$	8= $17+(-9)$
10= $3+7$	10= $5+5$	10= $7+3$	10= $9+1$	10= $11+(-1)$	10= $13+(-3)$	10= $15+(-5)$	10= $17+(-7)$
12= $3+9$	12= $5+7$	12= $7+5$	12= $9+3$	12= $11+1$	12= $13+(-1)$	12= $15+(-3)$	12= $17+(-5)$
14= $3+11$	14= $5+9$	14= $7+7$	14= $9+5$	14= $11+3$	14= $13+1$	14= $15+(-1)$	14= $17+(-3)$
16= $3+13$	16= $5+11$	16= $7+9$	16= $9+7$	16= $11+5$	16= $13+3$	16= $15+1$	16= $17+(-1)$
18= $3+15$	18= $5+13$	18= $7+11$	18= $9+9$	18= $11+7$	18= $13+5$	18= $15+3$	18= $17+1$
20= $3+17$	20= $5+15$	20= $7+13$	20= $9+11$	20= $11+9$	20= $13+7$	20= $15+5$	20= $17+3$
22= $3+19$	22= $5+17$	22= $7+15$	22= $9+13$	22= $11+11$	22= $13+9$	22= $15+7$	22= $17+5$
24= $3+21$	24= $5+19$	24= $7+17$	24= $9+15$	24= $11+13$	24= $13+11$	24= $15+9$	24= $17+7$

FIGURE 1 : les énoncés vrais de la forme $pair = impair + impair$

Intéressons-nous maintenant à la conjecture de Goldbach. Quand on représente la comète des décompositions, que représentent les points de la comète ? (admettons qu'ils soient positionnés eux-aussi aux points entiers d'un maillage $\mathbb{N} \times \mathbb{N}$).



Un point (x, y) bleu de la comète représente qu'on a y énoncés vrais de la forme '$x = p + q$' avec x pair, p et q deux nombres premiers (et $p \leqslant q$ en l'occurrence). Par exemple, pour le nombre 98, qui a 3 décompositions de Goldbach 19+79, 31+67 et 37+61, le point de la comète $(98, 3)$ représente le fait que parmi les multiples possibilités d'énoncés en logique propositionnelle représentant les décompositions de 98 en une somme de deux impairs, 3 seulement font intervenir deux nombres premiers.

Il faut avoir à l'esprit que le point $(98, 3)$ représente les décompositions de Goldbach de 98 parmi une multitude d'autres points de la verticale $x = 98$, sur laquelle on pourrait positionner une multitude d'autres points qui pourraient signifier par exemple '98 a 10 décompositions de Goldbach', voire '98 a $\sqrt{2}$ décompositions de Goldbach' (si l'on se place dans $\mathbb{R}^2$ plutôt que dans $\mathbb{N}^2$).

L'article [1] fournit une limite asymptotique à la proportion des tautologies à une variable parmi les propositions à une variable. Une reformulation d'un extrait du résumé de cet article est :

> "*On démontre que la limite de la densité (ou bien la probabilité asymptotique) des formules propositionnelles intuitionnistes prouvables parmi l'ensemble de toutes les formules existe et qu'elle est égale, pour les types avec un type de base (les formules avec une seule variable propositionnelle), à $1/2 + \sqrt{5}/10$, qui est approximativement égal à 72%. Cela signifie qu'un type aléatoire (en terme de formule) d'une grande taille a environ 72% de chance d'être une tautologie.*"

Ce ratio est égal à $\dfrac{1}{2}+\dfrac{\sqrt{5}}{10} = 0.72360679775 = \dfrac{1+\sqrt{5}}{2\sqrt{5}}$, le nombre d'or divisé par $\sqrt{5}$.

Si l'on se dit que les points de la comète représentent un certain nombre de tautologies parmi de multiples propositions possibles, le ratio en question devrait minorer la comète.

C'est ce qui est montré par la courbe rouge, sous la comète, qui est le graphique de la fonction $f(x) = \dfrac{1+\sqrt{5}}{2\sqrt{5}}\ \dfrac{x}{\ln x \cdot \ln x}$.

On a un problème par rapport à l'interprétation des termes du résumé "*un type aléatoire (en terme de formule)* d'une grande taille" : tous nos énoncés de la forme $pair = impair + impair$ sont de la même taille (on n'a pas d'énoncés faisant intervenir de multiples implications, dussent-elles porter sur une et une seule variable). On pourrait considérer que l'énoncé '100=3+97' est d'une plus grande taille que l'énoncé '6=3+3' parce que, calculatoirement parlant, le calcul est plus long à effectuer (démontrer par ordinateur que 97 est premier nécessite plus de temps que démontrer que 3 l'est) mais on pensait plutôt qu'on devait se situer là dans une sorte de monde idéal, où tous les énoncés de la forme 'tel nombre est premier' sont donnés "d'un coup". On a donc du mal à savoir si (et via) quelle transposition logique de la conjecture de Goldbach dans la logique propositionnelle des énoncés à une seule variable le résultat de [1] est applicable.

Cependant, si l'interprétation de la comète et la transposition qu'on a proposée étaient valides, on aurait établi un joli "pont", au sens d'Olivia Caramello, entre l'arithmétique et la logique propositionnelle à une variable.

Note : on a déposé dans un fichier tableur de 25 méga les nombres de décompositions de Goldbach des nombres pairs de 6 à 10^6 ainsi que la valeur de la fonction minorante rouge, à l'adresse http://denise.vella.chemla.free.fr/dg-calc-1000000-dvc.pdf[1].

Références

[1] Malgorzata Moczurad, Jerzy Tyszkiewicz, Marek Zaionc, *Statistical properties of simple types.* Math. Struct. Comput. Sci. 10(5): 575-594 (2000).
Traduction en français : http://denise.vella.chemla.free.fr/trad-Prop-stat-tautologies.pdf.
Article original au format pdf (le fichier au format postscript est téléchargeable sur la page de M. Zaionk).
http://denise.vella.chemla.free.fr/Statistical$_Properties_of_Simple_Types.pdf$.

[2] Olivia Caramello, *Theories, Sites, Toposes: Relating and studying mathematical theories through topos-theoretic 'bridges'*, Oxford University Press, 2017.

---

[1] Il y a 32 exceptions à la minoration pour les pairs inférieurs à 10^6. Ces exceptions concernent les nombres : $6 = 2\cdot 3, 8 = 2^3, 12 = 2^2\cdot 3, 38 = 2\cdot 19, 68 = 2^2\cdot 17, 98 = 2\cdot 7^2, 128 = 2^7, 152 = 2^3\cdot 19, 326 = 2\cdot 163, 332 = 2^2\cdot 183, 398 = 2\cdot 199, 488 = 2^3\cdot 61, 632 = 2^3\cdot 79, 668 = 2^2\cdot 167, 692 = 2^2\cdot 173, 992 = 2^5\cdot 31, 1112 = 2^3\cdot 139, 1412 = 2^2\cdot 353, 1718 = 2\cdot 859, 2048 = 2^{11}, 2252 = 2^2\cdot 563, 2642 = 2\cdot 1321, 2672 = 2^4\cdot 167, 2936 = 2^3\cdot 367, 4412 = 2^2\cdot 1103, 5468 = 2^2\cdot 1367, 5948 = 2^2\cdot 1487, 7508 = 2^2\cdot 1877, 8042 = 2\cdot 4021, 8048 = 2^4\cdot 503, 8552 = 2^3\cdot 1069, 9602 = 2\cdot 4801$.

Probabilité d'obtenir une décomposition de Goldbach d'un nombre pair (Denise Vella-Chemla, août 2022)

1. Un exemple illustratif

Prenons l'exemple de la recherche des décomposants de Goldbach de l'entier pair $n = 98$.

$$S_{98} = \begin{cases} 98 \equiv 0 \pmod 2 \\ 98 \equiv 2 \pmod 3 \\ 98 \equiv 3 \pmod 5 \\ 98 \equiv 0 \pmod 7 \end{cases}$$

Appelons d_{98} un décomposant de Goldbach potentiel de $n = 98$. d_{98} peut être congru, hormis 0, à tout ce à quoi $n = 98$ n'est pas congru. Le signe $\vee$ dans le système ci-dessous est à lire comme un ou exclusif, son emploi étendu est à comprendre comme le fait de vérifier autant de systèmes de congruences que la combinatoire le permet.

$$S_{d_{98}} = \begin{cases} d_{98} \equiv 1 \pmod 2 \\ d_{98} \equiv 1 \pmod 3 \\ d_{98} \equiv 1 \vee 2 \vee 4 \pmod 5 \\ d_{98} \equiv 1 \vee 2 \vee 3 \vee 4 \vee 5 \vee 6 \pmod 7 \end{cases}$$

Remarque : on note que le fait de respecter le système de systèmes de congruences ci-dessus est une condition suffisante mais non nécessaire pour obtenir un décomposant de Goldbach de n. La démonstration de la validité de cette caractérisation des décomposants de Goldbach d'un nombre pair n qui sont supérieurs à la racine carrée de n est fournie en section 2.

Comme on le comprend aisément, les modules qui ne divisent pas n "éliminent davantage de classes de congruences" (au nombre de 2 par module premier inférieur à $\sqrt{n}$) que les modules qui divisent n. Plaçons-nous dans le pire des cas, où l'on élimine deux classes de congruences par module premier inférieur à $\sqrt{n}$, on trouve tout de même

$$\prod_{\substack{p \text{ premier} \\ 3 \leq p \leq \sqrt{n}}} (p-2)$$

classes de congruences *différentes* par l'application du théorème des restes chinois à chacun des systèmes de congruences combinatoirement trouvé (voir $S_{d_{98}}$ ci-dessus). Ces solutions sont inférieures à $D = \prod_{\substack{p \text{ premier} \\ 3 \leq p \leq \sqrt{n}}} p$.

Serait-il possible de "rater l'intervalle visé", i.e. que toutes les solutions soient supérieures à n, comprises entre n et D ? À la section 3, on verra que la probabilité d'obtenir au moins une solution inférieure à n tend très vite vers 1.

2. Caractérisation des décomposants de Goldbach de n supérieurs à $\sqrt{n}$[1]

Soit $n \in 2\mathbb{N} + 6$ un entier pair supérieur à 6.

[1]Leila Schneps a démontré que la caractérisation de certains décomposants de Goldbach proposée était valide.

Pour tout $p \in \mathbb{P}^*$ premier impair inférieur à $\sqrt{n}$ (*i.e.* $3 \leq p \leq \sqrt{n}$), on définit l'ensemble :

$$F_n(p) = \{m \in 2\mathbb{N}+1 : 3 \leq m \leq n/2,\ m \not\equiv 0\ [p],\ m \not\equiv n\ [p]\}$$

L'intersection des ensembles $F_n(p)$ pour tout p premier compris entre 3 et $\sqrt{n}$ est notée :

$$D_n = \bigcap_{\substack{p\in\mathbb{P} \\ 3\leq p\leq\sqrt{n}}} F_n(p)$$

Nous allons montrer que D_n et son complémentaire $n-D_n$ ne contiennent que des nombres premiers.
Lemme 1 : Soit $m \in 2\mathbb{N}+1$ un entier impair. Si m n'est divisible par aucun nombre premier compris entre 3 et $\sqrt{m}$, alors il est premier.

Démonstration : Si m est composé, on a $m = pq$, où p est le plus petit nombre premier intervenant dans la factorisation de m en nombres premiers et où q est le produit de tous les autres facteurs. Puisque m est impair, $p \geq 3$, et puisque $q \geq p$ (q étant le produit d'entiers $\geq p$), $m = pq \geq pp = p^2$ et donc $\sqrt{m} \geq p$ (la fonction racine carrée étant croissante). On a ainsi montré que si m impair est composé, il est divisible par un premier compris entre 3 et $\sqrt{m}$. Le lemme s'obtient par contraposition. $\square$

Lemme 2 : $D_n \subseteq \mathbb{P}$

Démonstration : Soit $m \in D_n$. Alors $m \in F_n(p)$ pour tout p premier compris entre 3 et $\sqrt{n}$. Par conséquent, m est impair et m n'est divisible par aucun nombre premier p compris entre 3 et $\sqrt{n}$ (puisque $m \not\equiv 0\ [p]$), et donc *a fortiori* par aucun premier compris entre 3 et $\sqrt{m}$ (car $m \leq n/2 \implies m \leq n \implies \sqrt{m} \leq \sqrt{n}$). D'après le lemme 1, m est donc premier. $\square$

Lemme 3 : $n - D_n \subseteq \mathbb{P}$

Démonstration : Soit $m \in D_n$. Alors $m \in F_n(p)$ pour tout p premier compris entre 3 et $\sqrt{n}$. Par conséquent, $n-m$ est impair (car m est impair et n pair) et $n-m$ n'est divisible par aucun nombre premier p compris entre 3 et $\sqrt{n}$ (puisque $m \not\equiv n\ [p]$), et donc *a fortiori* par aucun premier compris entre 3 et $\sqrt{n-m}$ (car $n-m \leq n \implies \sqrt{n-m} \leq \sqrt{n}$). D'après le lemme 1, $n-m$ est donc premier. $\square$

Les ensembles D_n ne contiennent que des décomposants de Goldbach de n.

Lemme 4 : Soit $n \in 2\mathbb{N}+6$. Si $D_n \neq \varnothing$, alors n vérifie la conjecture de Goldbach.

Démonstration : Si $D_n \neq \varnothing$, il contient un entier p nécessairement premier (d'après le lemme 1), tel que $q = n-p$ est également premier (d'après le lemme 2), et donc $n = p+q$ vérifie la conjecture de Goldbach.

3. Probabilité $P(n,k,p)$ de tirer un nombre inférieur ou égal à k, sans remise, quand on tire uniformément p entiers parmi les n premiers entiers.

La probabilité[2] $P(n,k,p)$ de tirer tirer un nombre inférieur ou égal à k, sans remise, quand on tire uniformément p entiers parmi les n premiers entiers se calcule par la formule suivante :

$$P \;= \frac{k}{n} + \frac{n-k}{n}\left(\frac{k}{n-1} + \frac{n-k-1}{n-1}\left(\frac{k}{n-2} + \frac{n-k-2}{n-2}\left(\ldots\left(\frac{k}{n-p+1}\right)\ldots\right)\right)\right)$$

Le premier terme de la somme correspond au fait de trouver un nombre inférieur à k dès le premier tirage. Le deuxième terme de la somme correspond au fait d'avoir tiré un nombre supérieur à k lors du premier tirage, de ne pas avoir la possibilité de tirer à nouveau ce nombre, et de tenter sa chance sur les nombres restant, la probabilité restant uniforme sur les nombres restant, etc.

On calcule cette probabilité pour

$$p = \prod_{\substack{x \text{ premier} \\ 3 \leqslant x \leqslant \sqrt{k}}} (x-2)$$

et

$$n = \prod_{\substack{x \text{ premier} \\ 3 \leqslant x \leqslant \sqrt{k}}} x.$$

Le programme python utilisé est le suivant :

```
import math

def P(n, k, p):
    assert(1 <= p and p <= n and k <= n-p)
    s, t = 0, 1
    for i in range(p):
        s += t*(k/(n-i))
        t *= (n-k-i)/(n-i)
    return s

for n, k, p in [(30, 26, 3),
                (210, 50, 15),
                (2310, 122, 135),
                (30030, 170, 1485),
                (510510, 290, 22275),
                (9699690, 362, 378675),
                (223092870, 530 , 7952175),
                (6469693230, 842, 214708725),
                (200560490130, 962, 6226553025)]:
    print(f'n = {n}, k = {k}, p = {p} : P_n(k,p) = {P(n, k, p)}')
```

```
n = 30, k = 26, p = 3 : P_n(k,p) = 0.9990147783251231
n = 210, k = 50, p = 15 : P_n(k,p) = 0.9856514594832753
n = 2310, k = 122, p = 135 : P_n(k,p) = 0.9994752040784769
n = 30030, k = 170, p = 1485 : P_n(k,p) = 0.999824267526177
n = 510510, k = 290, p = 22275 : P_n(k,p) = 0.9999976037996607
n = 9699690, k = 362, p = 378675 : P_n(k,p) = 0.9999994514468453
n = 223092870, k = 530, p = 7952175 : P_n(k,p) = 0.9999999955788792
n = 6469693230, k = 842, p = 214708725 : P_n(k,p) = 0.9999999997119475
n = 200560490130, k = 962, p = 6226553025 : P_n(k,p) = 0.9999999921336346
```

[2] Merci Jacques.

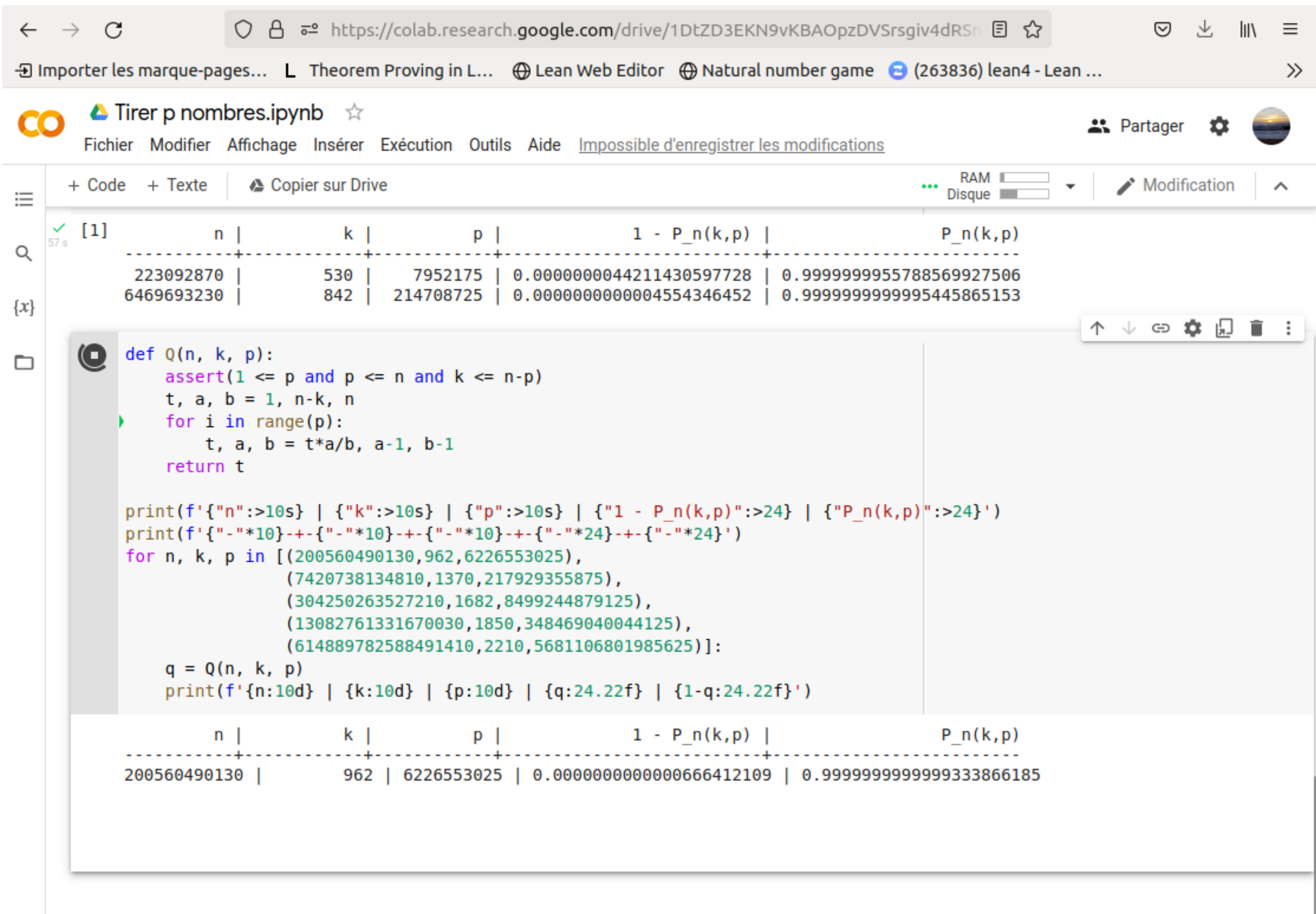
Fournissons ses résultats dans le tableau ci-dessous :

k	n	k^2+1	p	$P(n,k,p)$
5	30	26	3	0.9990147783251231
7	210	50	15	0.9856514594832753
11	2310	122	135	0.9994752040784769
13	30030	170	1485	0.999824267526177
17	510510	290	22275	0.9999976037996607
19	9699690	362	378675	0.9999994514468453
23	223092870	530	7952175	0.9999999955788792
29	6469693230	842	214708725	0.9999999997119475
31	200560490130	962	6226553025	0.9999999921336346

Pour confirmer les résultats du programme, on utilise une fonction qui calcule la probabilité des événements complémentaires, i.e. la probabilité qu'au cours des p tirages sans remise réalisés selon la loi uniforme discrète dans l'intervalle $1..n$, tous les entiers tirés soient strictement supérieurs à k ; on programme aussi ces fonctions en C++ pour comparer les résultats et essayer de gagner en rapidité. Malheureusement, la taille des nombres et donc le temps d'exécution nécessaire oblige à se contenter des probabilités jusqu'au nombre premier 37. Un compilateur C++ écrit une probabilité de 1 à partir du nombre premier 37.

```
emacs27@denise-Inspiron-3501
File  Edit  Options  Buffers  Tools  C++  Help
Save  Undo

#include <iostream>
#include <cmath>

long double Q(long double n, long double k, long double p) {
  long double t, i, a, b ;

  if ((p <= 0) or (p > n) or (k > n-p))
    std::cout << "erreur\n" ;
  else {
    t = 1 ;
    a = n-k ;
    b = n ;
    for (i = 0 ; i < p ; ++i) {
      t *= a/b ;
      a = a-1 ;
      b = b-1 ;
    }
    return 1-t ;
  }
}

int main (int argc, char* argv[])
{
  long double n, k, p ;

  n = 304250263527210 ; k = 1682 ; p = 8499244879125 ;
  std::cout << "n = " << n << " k = " << k << " p = " << p << " P_n(k,p) = " ;
  printf("%17.15Lf \n",Q(n,k,p)) ;
}

-:**-  prob.cpp      All L28     (C++//l Abbrev)
```

```python
def Q(n, p, k):
    assert(1 <= p and p <= n and k <= n-p)
    t, a, b = 1, n-k, n
    for i in range(p):
        t, a, b = t*a/b, a-1, b-1
    return t

print(f'{"n":>10s} | {"p":>10s} | {"k":>10s} | {"1 - P(n,p,k)":>24} | {"P(n,p,k)":>24}')
print(f'{"-"*10}-+-{"-"*10}-+-{"-"*10}-+-{"-"*24}-+-{"-"*24}')
for n, p, k in [(30, 3, 26), (210, 15, 50), (2310, 135, 122), (30030, 1485, 170), (510510, 22275, 290), (9699690, 378675, 362)]:
    q = Q(n, p, k)
    print(f'{n:10d} | {p:10d} | {k:10d} | {q:24.22f} | {1-q:24.22f}')
```




```
         n |          k |          p |             1 - P_n(k,p) |                 P_n(k,p)
-----------+------------+------------+--------------------------+-------------------------
 223092870 |        530 |    7952175 | 0.0000000044211430597728 | 0.9999999955788569927506
6469693230 |        842 |  214708725 | 0.0000000000004554346452 | 0.9999999999995445865153
```

```python
def Q(n, k, p):
    assert(1 <= p and p <= n and k <= n-p)
    t, a, b = 1, n-k, n
    for i in range(p):
        t, a, b = t*a/b, a-1, b-1
    return t

print(f'{"n":>10s} | {"k":>10s} | {"p":>10s} | {"1 - P_n(k,p)":>24} | {"P_n(k,p)":>24}')
print(f'{"-"*10}-+-{"-"*10}-+-{"-"*10}-+-{"-"*24}-+-{"-"*24}')
for n, k, p in [(200560490130,962,6226553025),
                (7420738134810,1370,217929355875),
                (304250263527210,1682,8499244879125),
                (13082761331670030,1850,348469040044125),
                (614889782588491410,2210,5681106801985625)]:
    q = Q(n, k, p)
    print(f'{n:10d} | {k:10d} | {p:10d} | {q:24.22f} | {1-q:24.22f}')
```

```
         n |          k |          p |             1 - P_n(k,p) |                 P_n(k,p)
-----------+------------+------------+--------------------------+-------------------------
200560490130 |        962 | 6226553025 | 0.0000000000000666412109 | 0.9999999999999333866185
```

```
denise@denise-Inspiron-3501:~/Bureau$ ./probasansdeux.exe
n = 30 k = 26 p = 3 P_n(k,p) = 0.999014778325123
n = 210 k = 50 p = 15 P_n(k,p) = 0.985651459483275
n = 210 k = 66 p = 15 P_n(k,p) = 0.997267239090670
n = 210 k = 80 p = 15 P_n(k,p) = 0.999458511995887
n = 210 k = 100 p = 15 P_n(k,p) = 0.999962342308524
n = 2310 k = 122 p = 135 P_n(k,p) = 0.999475204078478
n = 2310 k = 140 p = 135 P_n(k,p) = 0.999833990944990
n = 2310 k = 160 p = 135 P_n(k,p) = 0.999954324106302
n = 30030 k = 170 p = 1485 P_n(k,p) = 0.999824267526178
n = 30030 k = 220 p = 1485 P_n(k,p) = 0.999986315722772
n = 30030 k = 260 p = 1485 P_n(k,p) = 0.999998230122287
n = 510510 k = 290 p = 22275 P_n(k,p) = 0.999997603799665
n = 9.69969e+06 k = 362 p = 378675 P_n(k,p) = 0.999999451446880
n = 2.23093e+08 k = 530 p = 7.95218e+06 P_n(k,p) = 0.999999995578857
n = 6.46969e+09 k = 842 p = 2.14709e+08 P_n(k,p) = 0.999999999999545
```

```
denise@denise-Inspiron-3501:~/Bureau$ ./probasansdeux.exe
n = 7.42074e+12 k = 1370 p = 2.17929e+11 P_n(k,p) = 1.000000000000000
```

```
denise@denise-Inspiron-3501:~/Bureau$ ./probasansdeux.exe
n = 2.0056e+11 k = 962 p = 6.22655e+09 P_n(k,p) = 0.999999999999933
```

```
denise@denise-Inspiron-3501:~/Bureau$ ./probasansdeux.exe
n = 7.42074e+12 k = 1370 p = 2.17929e+11 P_n(k,p) = 1.000000000000000
```

Découverte d'une loi tout extraordinaire des nombres pairs par rapport à l'un de leur décomposant de Goldbach

Denise Vella-Chemla

Février 2023

On présente ici un résultat expérimental, qu'on a obtenu par hasard : on s'interrogeait sur la possibilité d'une relation entre tout décomposant de Goldbach d'un nombre pair et la moitié de ce nombre pair.

On n'a pas mis au jour une telle relation pour tous les décomposants de Goldbach, mais pour au moins l'un d'entre eux, et ce jusqu'à 10^9.

Pour les nombres pairs de la forme $4k$, on réussit toujours à trouver, à partir de $n = 8$ un nombre premier p qui est un décomposant de Goldbach de n, i.e. tel que p et $n - p$ sont premiers, mais également tel que $(n/2) + 2 - p$ est premier aussi.

Pour les nombres pairs de la forme $4k + 2$, on réussit toujours à trouver, à partir de $n = 10$ un nombre premier p qui est un décomposant de Goldbach de n, i.e. tel que p et $n - p$ sont premiers, mais également tel que $(n/2) + 1 - p$ est premier aussi.

On écrit dans deux tableaux (l'un contenant les nombres pairs de la forme $4k$, l'autre les nombres pairs de la forme $4k + 2$), à la manière d'Euler dans son article *Découverte d'une loi tout extraordinaire des nombres par rapport à la somme de leurs diviseurs*, les quadruplets de nombres, de la forme $(n, p_1, p_2, n - p_1)$ tels que p_1, $p_2 = (n/2) + \varepsilon - p_1$, $n - p_1$ sont tous les trois premiers, avec ε qui est égal à 2 ou 1 selon que n est le double d'un nombre pair (i.e. de la forme $4k$) ou le double d'un nombre impair (i.e. de la forme $4k + 2$).

Pour l'instant, on reste surprise. Le risque, lorsqu'on s'émerveille du résultat d'un programme, c'est de s'émerveiller alors que cela n'en vaut pas la peine[1].

On peut interpréter le tableau de gauche ci-dessous de la façon suivante : il semblerait (jusqu'à 10 milliards) qu'un nombre pair n de la forme $4k$ partage toujours l'un de ses décomposants de Goldbach avec sa moitié à laquelle on ajoute 2. Les premières lignes du tableau se lisent ainsi : 8 partage avec 6 le décomposant de Goldbach 3, puis (deuxième ligne), 12 partage avec 8 le décomposant de Goldbach 5, puis (troisième ligne), 16 partage avec 10 le décomposant de Goldbach 3, puis 20 partage avec 12 le décomposant de Goldbach 7...

Le tableau de droite s'interprète de la façon suivante : il semblerait (jusqu'à 40 milliards) qu'un nombre pair n de la forme $4k + 2$ partage toujours l'un de ses décomposants de Goldbach avec sa moitié à laquelle on ajoute 1. Les premières lignes du tableau se lisent ainsi : 10 partage avec 6 le

[1]Lire ces belles phrases de Maryam Mirzakhani pour décrire ce sentiment, à la page 16 de http://denisevellachemla.eu/trad-Maryam-Secrets-de-la-Surface.pdf, en espérant que la traduction ne trahit pas ce que dit d'elle son mari, Jan Vondrák : "Parfois, vous voyez, elle était heureuse lorsqu'elle disait "Oh j'ai cette idée, et j'espère qu'elle est correcte. En fait, permettez-moi de ne pas y penser pendant quelques heures avant de découvrir qu'elle est fausse... Mais je veux juste profiter du sentiment que cette idée *pourrait être correcte* !".

décomposant de Goldbach 3, puis (deuxième ligne), 14 partage avec 8 le décomposant de Goldbach 3, puis (troisième ligne), 18 partage avec 10 le décomposant de Goldbach 5, puis 22 partage avec 12 le décomposant de Goldbach 5...

C'est un peu comme si certaines décompositions de Goldbach étaient "envoyées" par les petits nombres pairs à des nombres pairs de plus en plus éloignés d'eux (éloignés d'eux selon la liste des nombres pairs successifs).

n	p_1	$((n/2)+2)-p_1$	$n-p_1$
8	3	$(4+2)-3=3$	5
12	5	$(6+2)-5=3$	7
16	3	$(8+2)-3=7$	13
20	7	$(10+2)-7=5$	13
24	7	$(12+2)-7=7$	17
28	5	$(14+2)-5=11$	23
32	13	$(16+2)-13=5$	19
36	7	$(18+2)-7=13$	29
40	3	$(20+2)-3=19$	37
44	7	$(22+2)-7=17$	37
48	7	$(24+2)-7=19$	41
52	5	$(26+2)-5=23$	47
56	13	$(28+2)-13=17$	43
60	13	$(30+2)-13=19$	47
64	3	$(32+2)-3=31$	61
68	7	$(34+2)-7=29$	61
72	19	$(36+2)-19=19$	53
76	3	$(38+2)-3=37$	73
80	13	$(40+2)-13=29$	67
84	13	$(42+2)-13=31$	71
88	5	$(44+2)-5=41$	83
92	19	$(46+2)-19=29$	73
96	7	$(48+2)-7=43$	89
100	11	$(50+2)-11=41$	89

Tableau 1 : Les nombres pairs de la forme $4k$

n	p_1	$((n/2)+1)-p_1$	$n-p_1$
10	3	$(5+1)-3=3$	7
14	3	$(7+1)-3=5$	11
18	5	$(9+1)-5=5$	13
22	5	$(11+1)-5=7$	17
26	3	$(13+1)-3=11$	23
30	11	$(15+1)-11=5$	19
34	5	$(17+1)-5=13$	29
38	7	$(19+1)-7=13$	31
42	5	$(21+1)-5=17$	37
46	5	$(23+1)-5=19$	41
50	3	$(25+1)-3=23$	47
54	11	$(27+1)-11=17$	43
58	11	$(29+1)-11=19$	47
62	3	$(31+1)-3=29$	59
66	5	$(33+1)-5=29$	61
70	17	$(35+1)-17=19$	53
74	7	$(37+1)-7=31$	67
78	11	$(39+1)-11=29$	67
82	11	$(41+1)-11=31$	71
86	3	$(43+1)-3=41$	83
90	17	$(45+1)-17=29$	73
94	5	$(47+1)-5=43$	79
98	19	$(49+1)-19=31$	79

Tableau 2 : Les nombres pairs de la forme $4k+2$

En annexe, est fourni le programme de détermination d'un quadruplet pour tout nombre pair. On a pu le tester (ainsi que le programme concernant les $4k+2$, dans lequel ε est égal à 1 au lieu de 2) jusqu'à un certain nombre de milliards.

On a enfin testé la même idée pour les nombres de la forme 4.10^k jusqu'à 40 milliards ou pour les nombres 10×9^k ou bien 10×11^k. (Pourquoi d'une part, des multiplications successives par 9 et d'autre part, des multiplications successives par 11 ? Simplement pour "rester dans les $4k+2$" (des multiplications successives par 10 feraient passer une fois sur deux d'un nombre de la forme $4k$ à un nombre de la forme $4k+2$ et inversement les fois intermédiaires).

Multiplications par 10 (à partir de 4) :

n	p_1	$((n/2)+2)-p_1$	$n-p_1$
40	3	$(20+2)-3=19$	37
400	3	$(200+2)-3=199$	397
4 000	53	$(2\,000+2)-53=1\,949$	3 947
40 000	11	$(20\,000+2)-11=19\,991$	39 989
400 000	113	$(200\,000+2)-113=199\,889$	399 887
4 000 000	149	$(2\,000\,000+2)-149=1\,999\,853$	3 999 951
40 000 000	563	$(20\,000\,000+2)-563=19\,999\,439$	39 999 437
400 000 000	53	$(200\,000\,000+2)-53=199\,999\,949$	399 999 947
4 000 000 000	89	$(2\,000\,000\,000+2)-89=19\,999\,999\,913$	3 999 999 911
40 000 000 000	641	$(20\,000\,000\,000+2)-641=19\,999\,999\,361$	39 999 999 359

Multiplications par 9 (à partir de 10) :

n	p_1	$((n/2)+1)-p_1$	$n-p_1$
10	3	$(5+1)-3=3$	7
90	17	$(45+1)-17=29$	73
810	23	$(405+1)-23=383$	787
7 290	53	$(3\,645+1)-53=3\,593$	7 237
65 610	23	$(32\,805+1)-23=32\,783$	65 587
590 490	59	$(295\,245+1)-59=295\,187$	590 431
5 314 410	659	$(2\,657\,205+1)-659=2\,656\,547$	5 313 751
47 829 690	47	$(23\,914\,845+1)-47=23\,914\,799$	47 829 643
430 467 210	1103	$(215\,233\,605+1)-1\,103=215\,232\,503$	430 466 107
3 874 204 890	467	$(1\,937\,102\,445+1)-467=1\,937\,101\,979$	3 874 204 423

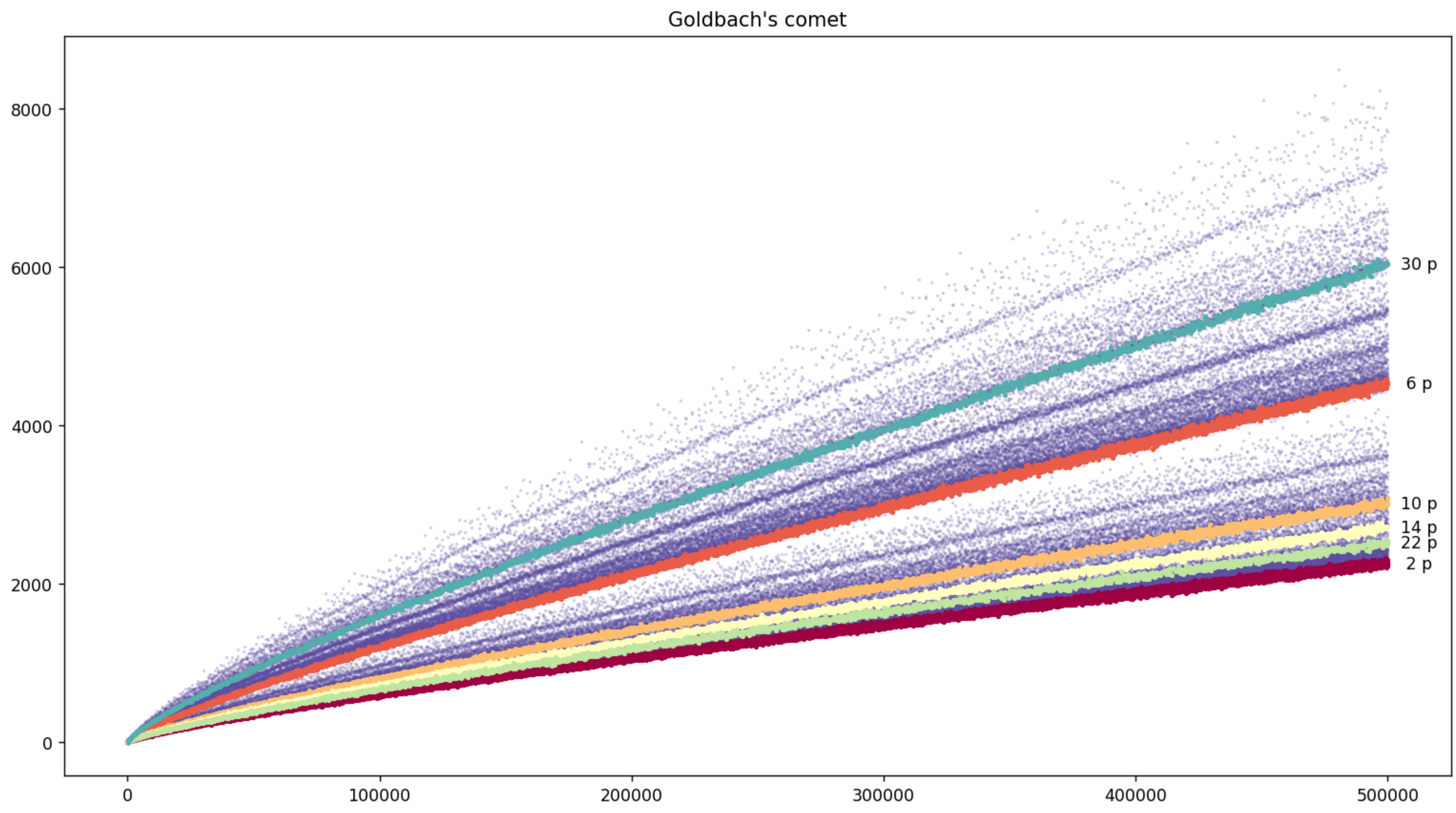
Multiplications par 11 (à partir de 10) :

n	p_1	$((n/2)+1)-p_1$	$n-p_1$
10	3	$(5+1)-3=3$	7
110	3	$(55+1)-3=53$	107
1 210	29	$(605+1)-29=577$	1 181
13 310	19	$(6\,655+1)-19=6\,637$	13 291
146 410	197	$(73\,205+1)-197=73\,009$	146 213
1 610 510	37	$(805\,255+1)-37=805\,219$	1610473
17 715 610	53	$(8\,857\,805+1)-53=8\,857\,753$	17 715 557
194 871 710	127	$(97\,435\,855+1)-127=97\,435\,729$	194 871 583
2 143 588 810	269	$(1\,071\,794\,405+1)-269=1\,071\,794\,137$	2 143 588 541

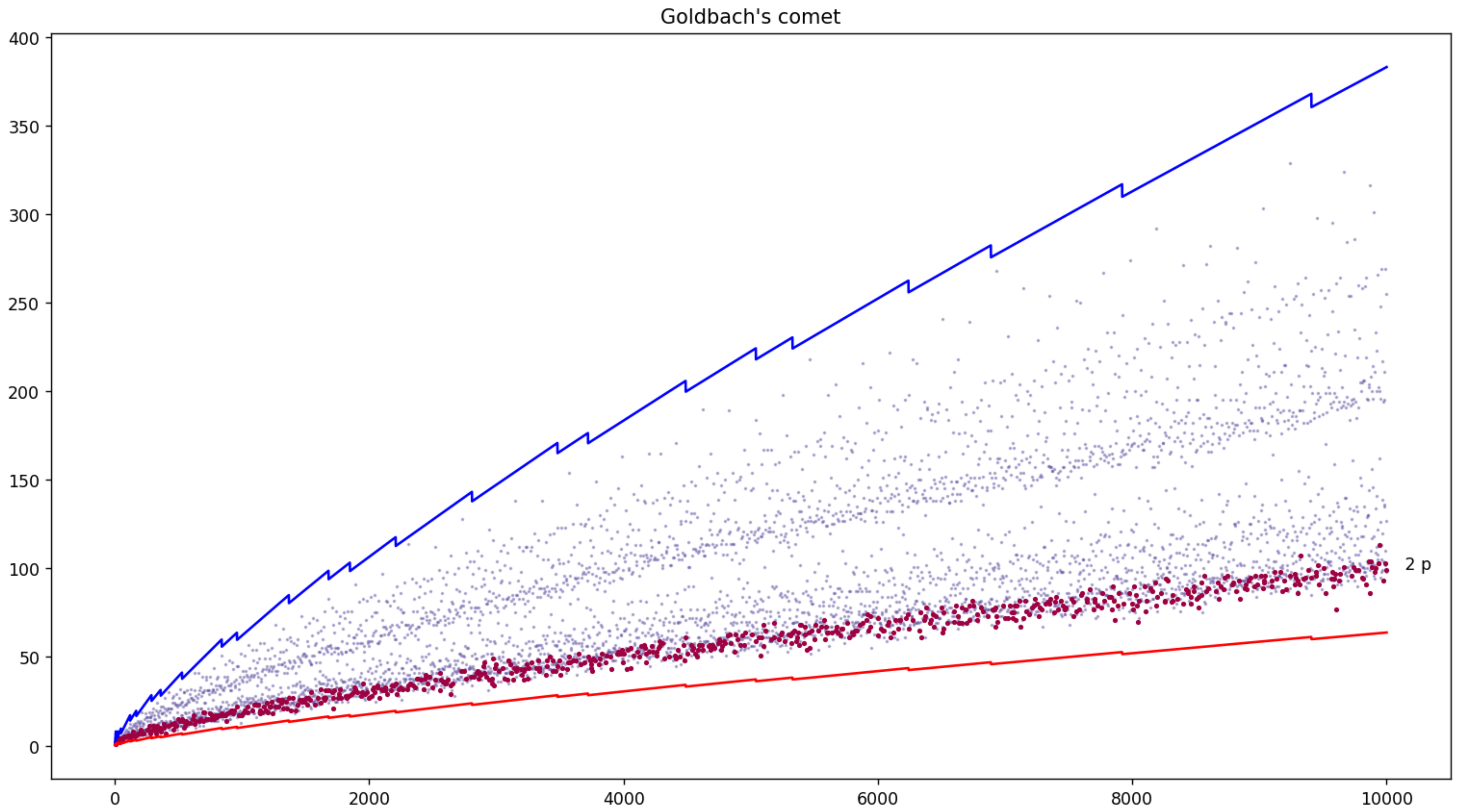
Y aurait-il une loi de récurrence, derrière certaines décompositions de Goldbach du moins ?

Annexe : Programme python de découverte de ces relations surprenantes

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

    def reverseList(L):
    lReverse = []
    for item in L:
        lReverse.insert(0, item)
    return lReverse

ok = []
fichier = open(”fic.txt”, ”a”)
for n in range(4,1000,4):
    trouve = False
    for p in range(3,int(n/2),2):
        if not trouve:
            if prime(p) and prime(n-p):
                if prime((n/2)-p+2):
                    print(’youpi’, n, p)
                    fichier.write(str(n))
                    fichier.write(”)
                    fichier.write(str(p))
                    fichier.write(”)
                    fichier.write(str(int((n/2)-p+2)))
                    fichier.write(”)
                    fichier.write(”)
                    trouve = True
                    ok.append(True)
    if not trouve:
        ok.append(False)
print(ok)
okrev = reverseList(ok)
print(okrev.index(False))
fichier.close()
```



Goldbach's comet
8000
6000
4000
2000
0
0
100000
200000
300000
400000
500000
30 p
6 p
10 p
14 p
22 p
2 p


Goldbach's comet
400
350
300
250
200
150
100
50
0
0
2000
4000
6000
8000
10000
2 p

```python
import math
from math import log, sqrt

lesx = []
lesy = []
lesymin = []
lesymax = []
for x in range(2,m+1):
    prod = 1.0
    rac = int(sqrt(x))
    for p in range(3,rac+1):
        if p in primes:
            prod = prod*(1-(2/p))
    lesx.append(x)
    lesymin.append(prod*x)
    lesymax.append(prod*x/6)
plt.plot(lesx,lesymin,color='blue')
plt.plot(lesx,lesymax,color='red')
plt.show()
```

Annexe 3 : L"'empilement" des valuations p-adiques"

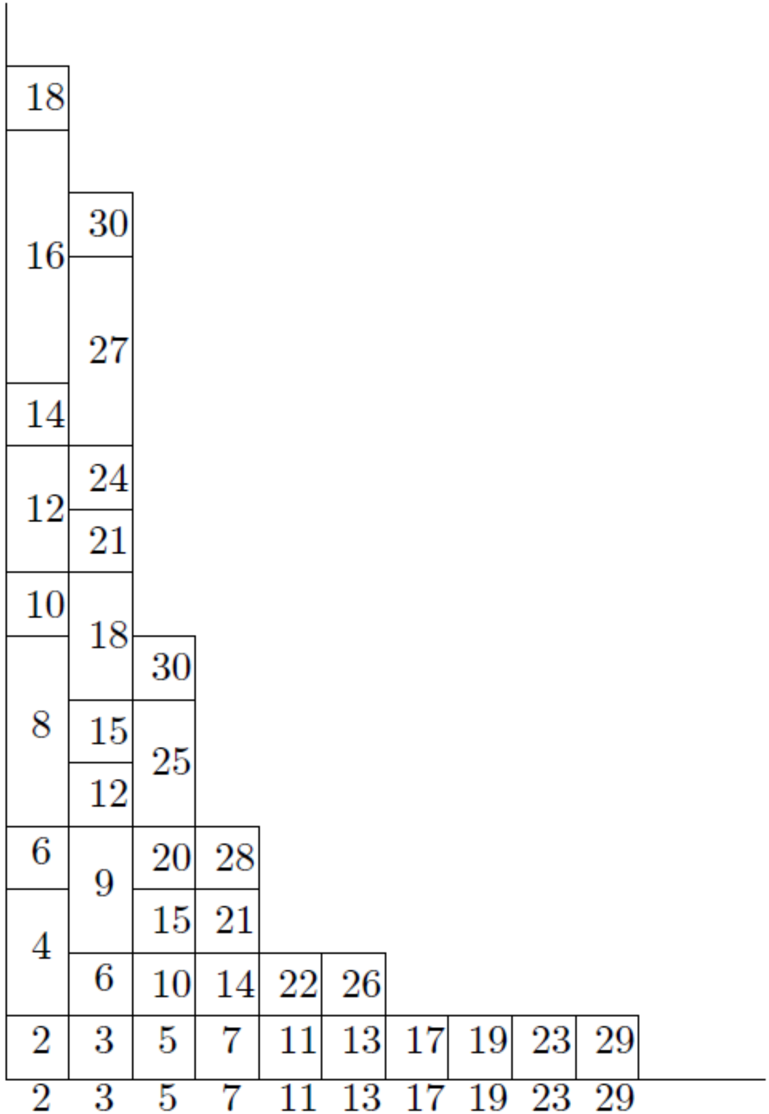


FIG. 4 – Courbe hyperbolique d'équation $xy = nlog(n)$

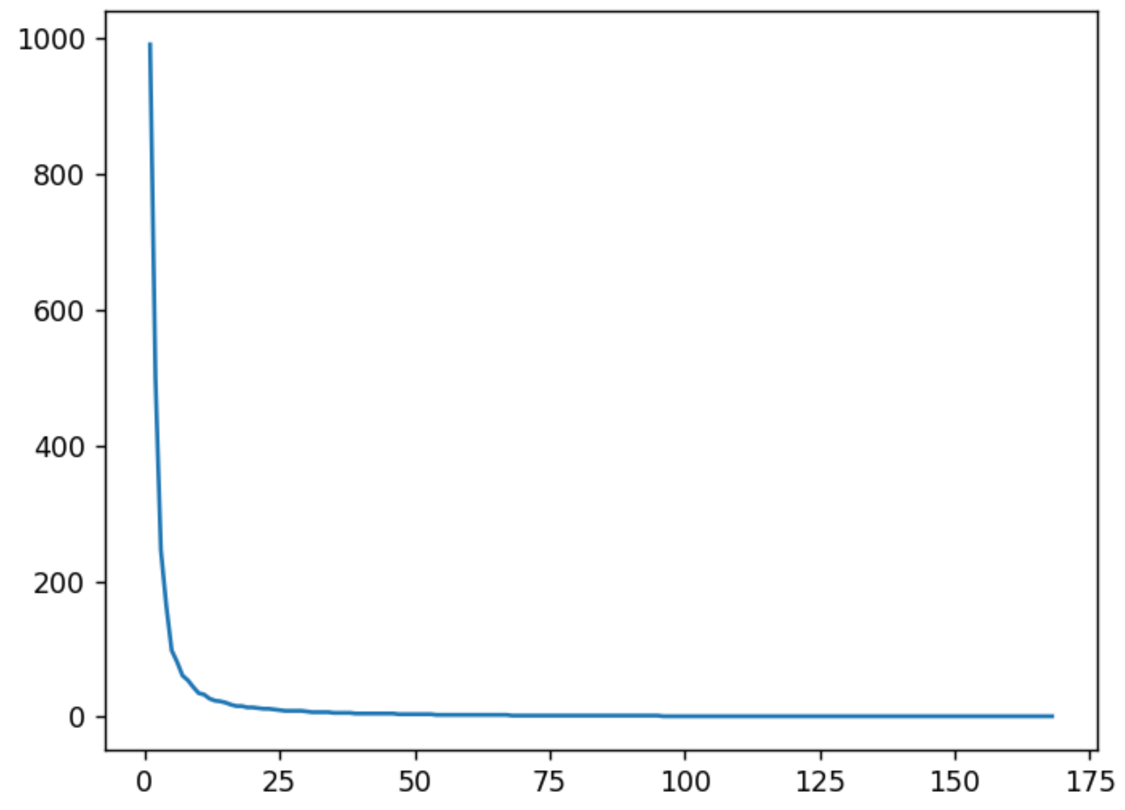
File Edit Selection View Go Run Terminal Help   rapidement-decroissante.py - Visual Studio Code

rapidement-decroissante.py

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > rapidement-decroissante.py > ...

```python
import matplotlib.pyplot as plt

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

def f(p,n):
    puiss = 1
    tempo = n ;
    while ((tempo/p > 0) and ((tempo % p) == 0)):
        tempo = tempo/p
        puiss = puiss+1
    return(puiss-1)

lesx = [] ; lesy = []
nbprem = 0 ; nbmax = 1000
for x in range(1, nbmax):
    if prime(x):
```

Figure 1



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER   Debug Console

```
Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\DENISE_2022\Desktop\tests-de-fonctions>  & 'C:\Users\DENISE_2022\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\DENISE_2022\.vscode\extensions\ms-python.python-2
022.16.1\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '57136' '--' 'c:\Users\DENISE_2022\Desktop\tests-de-fonctions\rapidement-decroissante.py'
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49,
50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96,
97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 1
35, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168] [991, 498, 246, 164
, 98, 81, 61, 54, 44, 35, 33, 27, 24, 23, 21, 18, 16, 16, 14, 14, 13, 12, 12, 11, 10, 9, 9, 9, 9, 8, 7, 7, 7, 7, 6, 6, 6, 6, 5, 5, 5, 5, 5, 5, 5, 5, 4, 4, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3,
3, 3, 3, 3, 3, 3, 3, 3, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
```

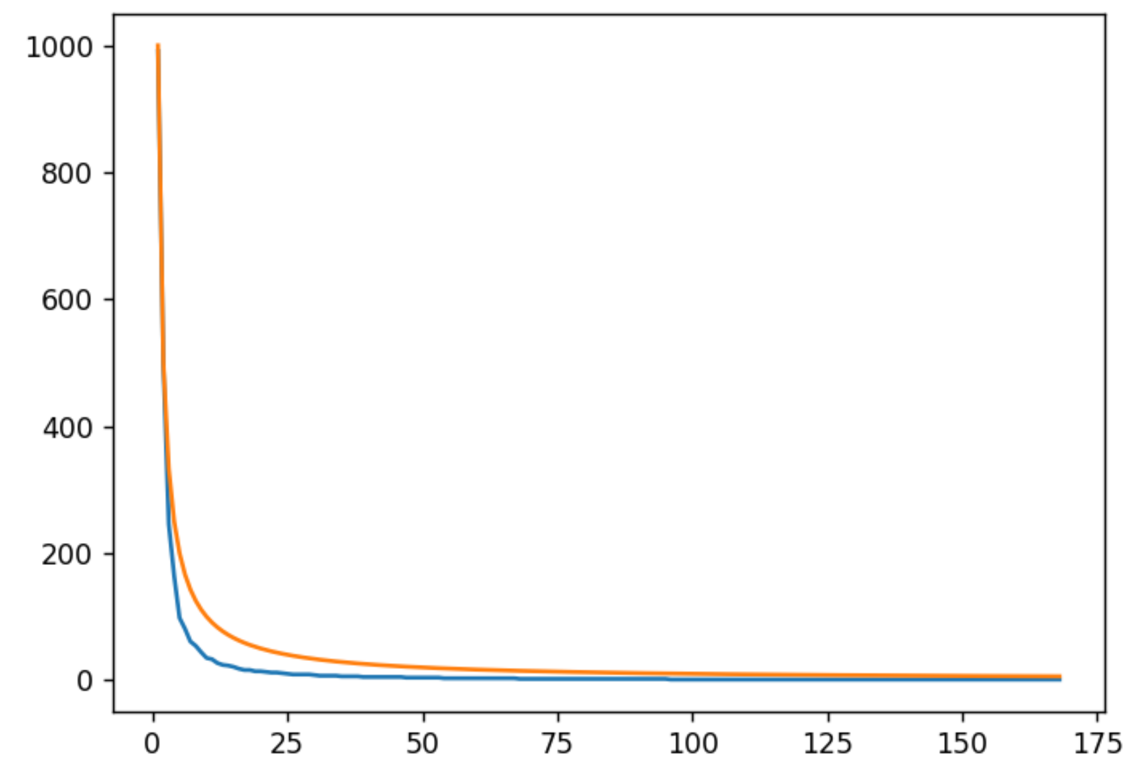
File Edit Selection View Go Run Terminal Help rapidement-decroissante-a-comparer.py - Visual Studio Code

rapidement-decroissante-a-comparer.py

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > rapidement-decroissante-a-comparer.py > ...

```python
  while ((tempo/p > 0) and ((tempo % p) == 0)):
    tempo = tempo/p
    puiss = puiss+1
  return(puiss-1)

lesx = [] ; lesy = [] ; lesyp = []
nbprem = 0 ; nbmax = 1000
for x in range(1, nbmax):
  if prime(x):
    nbprem = nbprem+1
    somme = 0
    for k in range(2, nbmax):
      somme = somme + f(x, k)
    lesx.append(nbprem)
    lesy.append(somme)
    lesyp.append(nbmax/nbprem)
print(lesy,lesyp)
plt.plot(lesx,lesy)
plt.plot(lesx,lesyp)
plt.show()
```

Figure 1



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
on-2022.16.1\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '59744' '--' 'c:\Users\DENISE_2022\Desktop\tests-de-fonctions\rapidement-decroissante-a-comparer.py'
[991, 498, 246, 164, 98, 81, 61, 54, 44, 35, 33, 27, 24, 23, 21, 18, 16, 16, 14, 14, 13, 12, 12, 11, 10, 9, 9, 9, 9, 8, 7, 7, 7, 7, 6, 6, 6, 6, 5, 5, 5, 5, 5, 5, 5, 5, 4, 4, 4, 4, 4, 4
, 4, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1] [1000.0, 500.0, 33
3.3333333333333, 250.0, 200.0, 166.66666666666666, 142.85714285714286, 125.0, 111.11111111111111, 100.0, 90.9090909090909, 83.33333333333333, 76.92307692307692, 71.42857142857143, 66.6
6666666666667, 62.5, 58.8235294117647, 55.55555555555556, 52.63157894736842, 50.0, 47.61904761904762, 45.45454545454545, 43.47826086956522, 41.666666666666664, 40.0, 38.46153846153846,
 37.03703703703704, 35.714285714285715, 34.48275862068966, 33.333333333333336, 32.25806451612903, 31.25, 30.303030303030305, 29.41176470588235, 28.571428571428573, 27.77777777777778, 2
7.027027027027028, 26.31578947368421, 25.641025641025642, 25.0, 24.390243902439025, 23.80952380952381, 23.25581395348837, 22.727272727272727, 22.22222222222222, 21.73913043478261, 21.2
7659574468085, 20.833333333333332, 20.408163265306122, 20.0, 19.607843137254903, 19.23076923076923, 18.867924528301888, 18.51851851851852, 18.181818181818183, 17.857142857142858, 17.54
385964912281, 17.24137931034483, 16.949152542372882, 16.666666666666668, 16.39344262295082, 16.129032258064516, 15.873015873015873, 15.625, 15.384615384615385, 15.151515151515152, 14.9
2537313432836, 14.705882352941176, 14.492753623188406, 14.285714285714286, 14.084507042253522, 13.88888888888889, 13.698630136986301, 13.513513513513514, 13.333333333333334, 13.157894
736842104, 12.987012987012987, 12.820512820512821, 12.658227848101266, 12.5, 12.345679012345679, 12.195121951219512, 12.048192771084338, 11.904761904761905, 11.764705882352942, 11.6279
06976744185, 11.494252873563218, 11.363636363636363, 11.235955056179776, 11.11111111111111, 10.989010989010989, 10.869565217391305, 10.75268817204301, 10.638297872340425, 10.5263157894
```

Python Deb...
Python Deb...

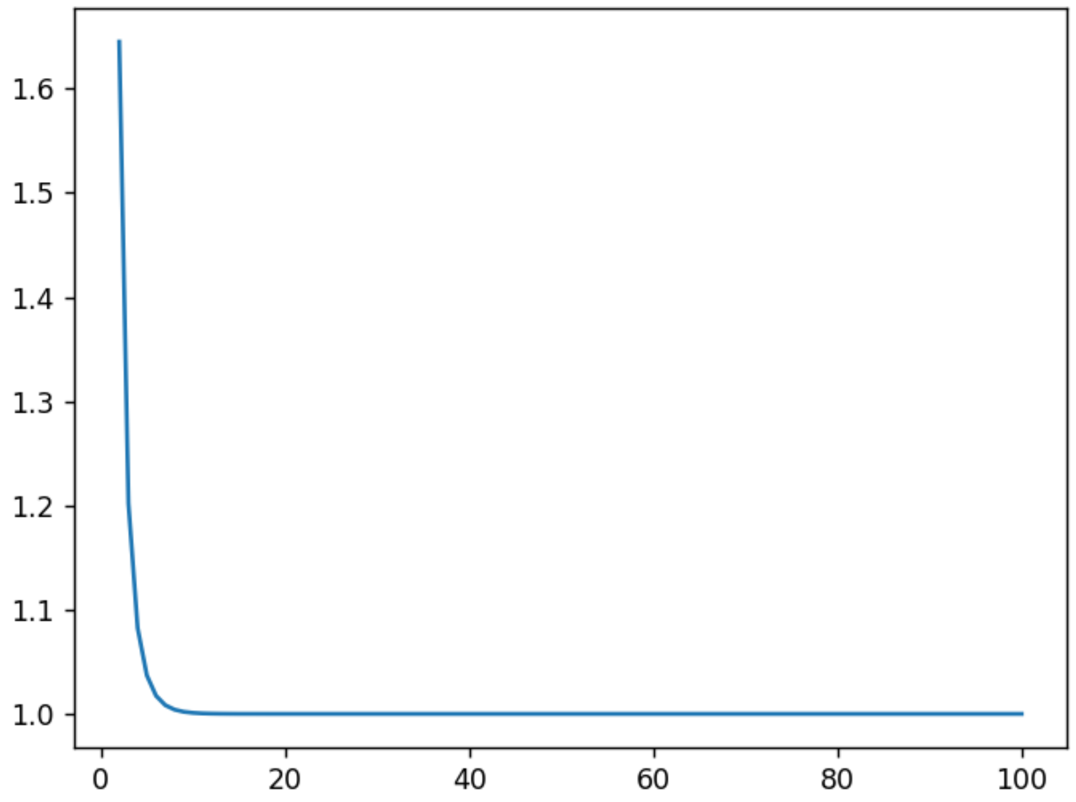
File Edit Selection View Go Run Terminal Help zetareel.py - Visual Studio Code

zetareel.py

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > zetareel.py > ...

```python
import scipy
from scipy.special import zeta
import matplotlib.pyplot as plt

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

lesx = []
lesy = []
for x in range(1,101):
    #if prime(x):
        lesx.append(x)
        y = zeta(x)
        lesy.append(y)
print(lesx)
print(lesy)
plt.plot(lesx, lesy)
```

Figure 1



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

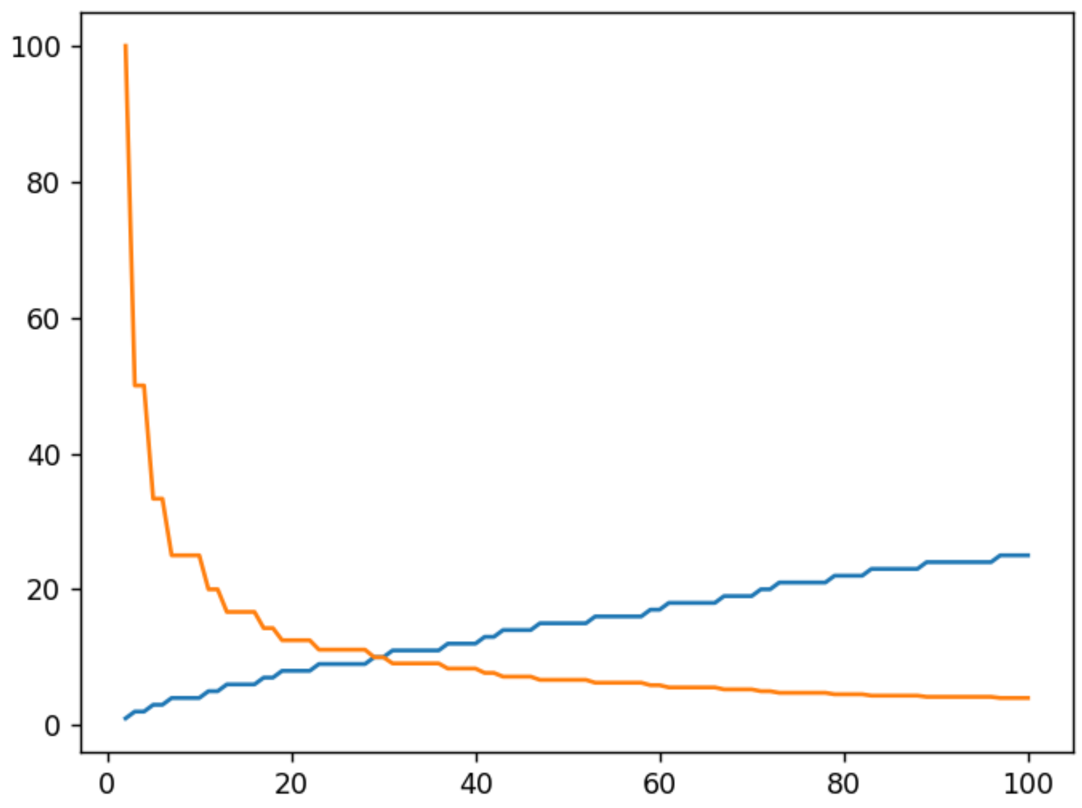
```
PS C:\Users\DENISE_2022\Desktop\tests-de-fonctions>  & 'C:\Users\DENISE_2022\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\DENISE_2022\.vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '57096' '--' 'c:\Users\DENISE_2022\Desktop\tests-de-fonctions\zetareel.py'
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]
[inf, 1.6449340668482264, 1.2020569031595942, 1.0823232337111381, 1.03692775514337, 1.0173430619844492, 1.008349277381923, 1.0040773561979444, 1.0020083928260821, 1.000994575127818, 1.0004941886041194, 1.000246086553308, 1.0001227133475785, 1.0000612481350588, 1.000030588236307, 1.0000152822594086, 1.0000076371976379, 1.000003817293265, 1.0000019082127165, 1.0000009539620338, 1.0000004769329869, 1.0000002384505027, 1.000000119219926, 1.000000059608189, 1.0000000298035034, 1.0000000149015549, 1.0000000074507118, 1.000000003725334, 1.0000000018626598, 1.0000000009313275, 1.0000000004656628, 1.0000000002328311, 1.0000000001164155, 1.0000000000582077, 1.0000000000291038, 1.000000000014552, 1.0000000000007276, 1.000000000003638, 1.000000000001819, 1.0000000000009095, 1.0000000000004547, 1.0000000000002274, 1.0000000000001137, 1.0000000000000568, 1.0000000000000284, 1.0000000000000142, 1.000000000000007, 1.0000000000000036, 1.0000000000000018, 1.000000000000009, 1.0000000000000004, 1.0000000000000002, 1.0000000000000002, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
```

toto.py

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > toto.py > ...

```python
  while True:
    if k * k > atester: return True
    else:
      if atester % k == 0: return False
      else: k = k + 1

def valpadic(p,x):
  exposant = 1
  tempo = x ;
  while ((tempo/p > 0) and ((tempo % p) == 0)):
      tempo = tempo/p
      exposant = exposant+1
  return(exposant-1)

pix = 0.0 ; xmax = 100 ; nbprem = 0
lesx = [] ; lesy = [] ; lesyp = []
for x in range(2,xmax+1):
  racx = int(sqrt(x))
  if prime(x):
    nbprem = nbprem+1
  somme = 1
  for p in range(2, racx+1):
    somme = somme + valpadic(p, x)
  pix = pix+floor(1.0/somme)
  lesx.append(x) ; lesy.append(pix) ; lesyp.append(xmax/nbprem)
  print(x, ' --> ', pix)
plt.plot(lesx,lesy)
plt.plot(lesx,lesyp)
print(lesx,lesy)
plt.show()
```

Figure 1



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Python Debug Console

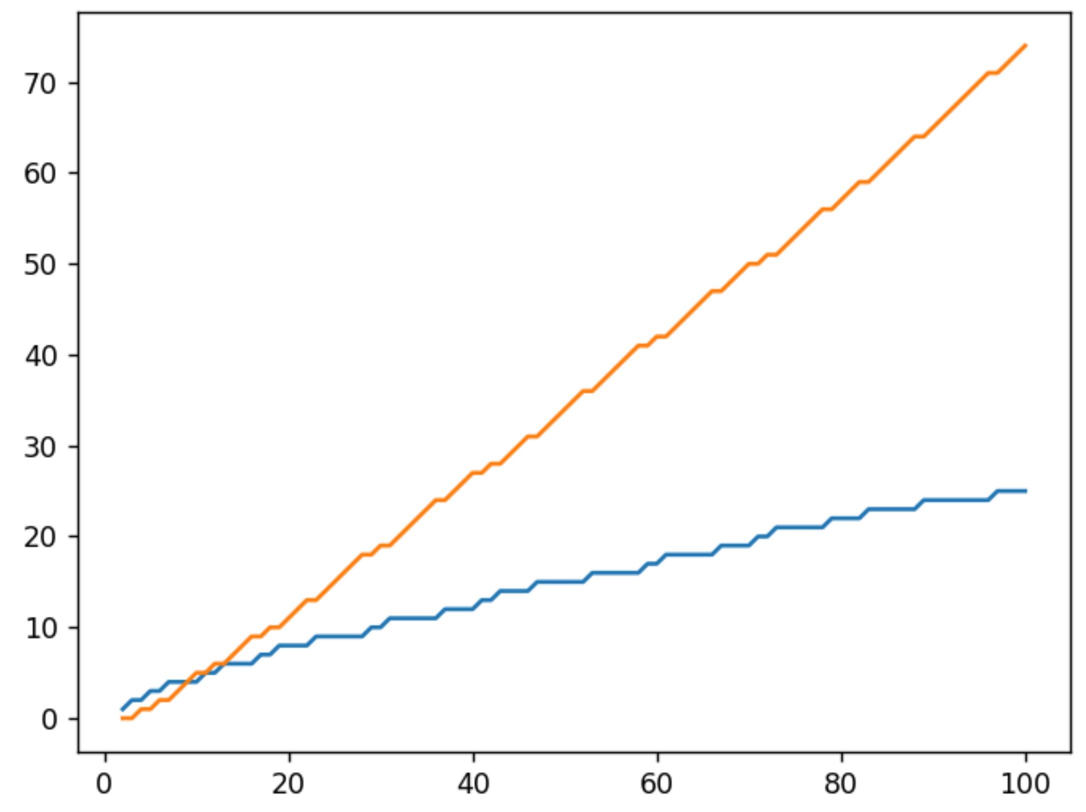
```
100  -->  25.0
[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53,
54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100] [1.0, 2.0, 2
.0, 3.0, 3.0, 4.0, 4.0, 4.0, 4.0, 5.0, 5.0, 6.0, 6.0, 6.0, 6.0, 7.0, 7.0, 8.0, 8.0, 8.0, 8.0, 9.0, 9.0, 9.0, 9.0, 9.0, 9.0, 10.0, 10.0, 11.0, 11.0, 11.0, 11.0, 11.0, 11.0, 12.0, 12.0, 12.0, 12.0, 13.0,
 13.0, 14.0, 14.0, 14.0, 14.0, 15.0, 15.0, 15.0, 15.0, 15.0, 15.0, 16.0, 16.0, 16.0, 16.0, 16.0, 16.0, 17.0, 17.0, 18.0, 18.0, 18.0, 18.0, 18.0, 18.0, 19.0, 19.0, 19.0, 19.0, 20.0, 20.0, 21.0, 21.0, 21
.0, 21.0, 21.0, 21.0, 22.0, 22.0, 22.0, 22.0, 23.0, 23.0, 23.0, 23.0, 23.0, 23.0, 24.0, 24.0, 24.0, 24.0, 24.0, 24.0, 24.0, 24.0, 25.0, 25.0, 25.0, 25.0]
```

toto.py

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > toto.py > ...

```python
def valpadic(p,x):
  exposant = 1
  tempo = x ;
  while ((tempo/p > 0) and ((tempo % p) == 0)):
      tempo = tempo/p
      exposant = exposant+1
  return(exposant-1)

pix = 0.0 ; pix2 = 0.0 ; xmax = 100 ; nbprem = 0
lesx = [] ; lesy = [] ; lesyp = []
for x in range(2,xmax+1):
  racx = int(sqrt(x))
  somme = 1
  somme2 = 1
  if prime(x):
    nbprem = nbprem+1
    somme2 = somme2 + (xmax/nbprem)
  for p in range(2, racx+1):
    somme = somme + valpadic(p, x)
  pix = pix + floor(1.0/somme)
  pix2 = pix2 + floor(1.0/somme2)
  lesx.append(x) ; lesy.append(pix) ; lesyp.append(pix2)
  print(x, ' --> ', pix)
plt.plot(lesx,lesy)
plt.plot(lesx,lesyp)
print(lesx,lesy,lesyp)
plt.show()
```

Figure 1



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

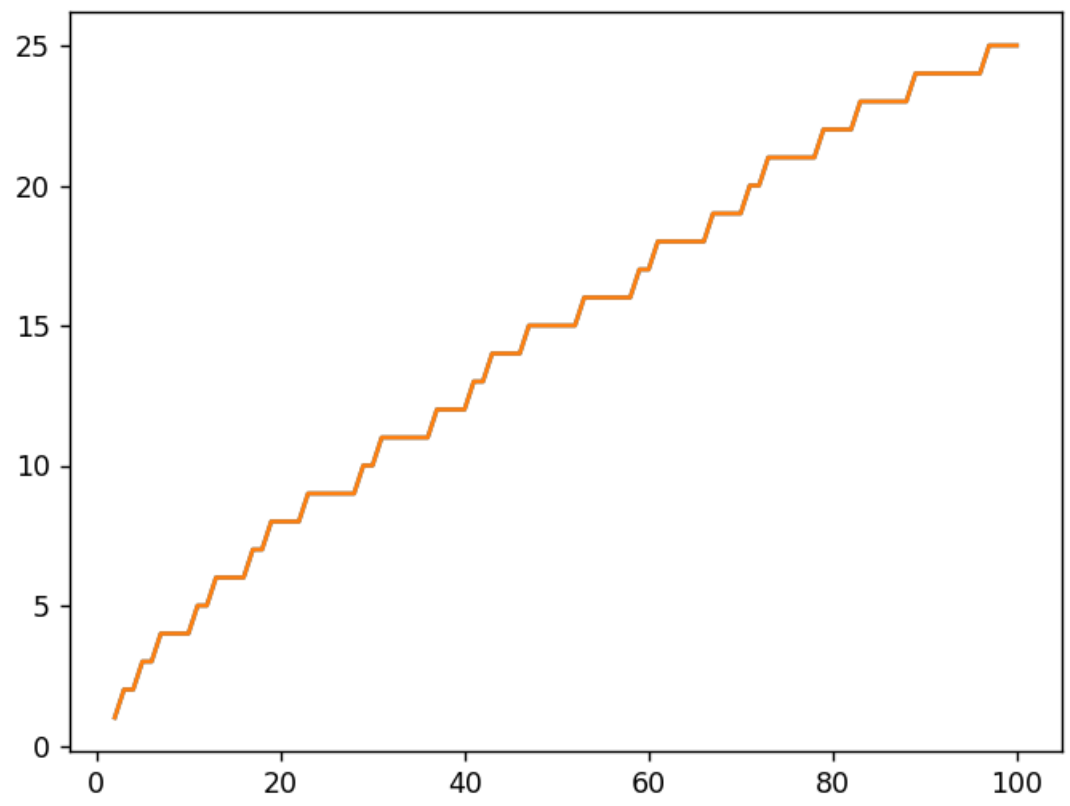
```
.0, 11.0, 11.0, 11.0, 12.0, 12.0, 12.0, 12.0, 13.0, 13.0, 14.0, 14.0, 14.0, 14.0, 15.0, 15.0, 15.0, 15.0, 15.0, 15.0, 16.0, 16.0, 16.0, 16.0, 16.0, 16.0, 17.0, 17.0, 18.0, 18.0, 18.0,
18.0, 18.0, 18.0, 19.0, 19.0, 19.0, 19.0, 20.0, 20.0, 21.0, 21.0, 21.0, 21.0, 21.0, 21.0, 22.0, 22.0, 22.0, 22.0, 23.0, 23.0, 23.0, 23.0, 23.0, 23.0, 24.0, 24.0, 24.0, 24.0, 24.0, 24.0
, 24.0, 24.0, 25.0, 25.0, 25.0, 25.0] [0.0, 0.0, 1.0, 1.0, 2.0, 2.0, 3.0, 4.0, 5.0, 5.0, 6.0, 6.0, 7.0, 8.0, 9.0, 9.0, 10.0, 10.0, 11.0, 12.0, 13.0, 13.0, 14.0, 15.0, 16.0, 17.0, 18.0,
 18.0, 19.0, 19.0, 20.0, 21.0, 22.0, 23.0, 24.0, 24.0, 25.0, 26.0, 27.0, 27.0, 28.0, 28.0, 29.0, 30.0, 31.0, 31.0, 32.0, 33.0, 34.0, 35.0, 36.0, 36.0, 37.0, 38.0, 39.0, 40.0, 41.0, 41.
0, 42.0, 42.0, 43.0, 44.0, 45.0, 46.0, 47.0, 47.0, 48.0, 49.0, 50.0, 50.0, 51.0, 51.0, 52.0, 53.0, 54.0, 55.0, 56.0, 56.0, 57.0, 58.0, 59.0, 59.0, 60.0, 61.0, 62.0, 63.0, 64.0, 64.0, 6
5.0, 66.0, 67.0, 68.0, 69.0, 70.0, 71.0, 71.0, 72.0, 73.0, 74.0]
```

toto.py ×

C: > Users > DENISE_2022 > Desktop > tests-de-fonctions > toto.py > ...

```python
def valpadic(p,x):
  exposant = 1
  tempo = x ;
  while ((tempo/p > 0) and ((tempo % p) == 0)):
      tempo = tempo/p
      exposant = exposant+1
  return(exposant-1)

pix = 0.0 ; pix2 = 0.0 ; xmax = 100 ; nbprem = 0
lesx = [] ; lesy = [] ; lesyp = []
for x in range(2,xmax+1):
  racx = int(sqrt(x))
  somme = 1
  somme2 = 1
  if prime(x):
    nbprem = nbprem+1
    somme2 = somme2 + (xmax/nbprem)
  for p in range(2, racx+1):
    somme = somme + valpadic(p, x)
  pix = pix + floor(1.0/somme)
  pix2 = pix2 + (1.0-floor(1.0/somme2))
  lesx.append(x) ; lesy.append(pix) ; lesyp.append(pix2)
  print(x, ' --> ', pix)
plt.plot(lesx,lesy)
plt.plot(lesx,lesyp)
print(lesx,lesy,lesyp)
plt.show()
```

Figure 1



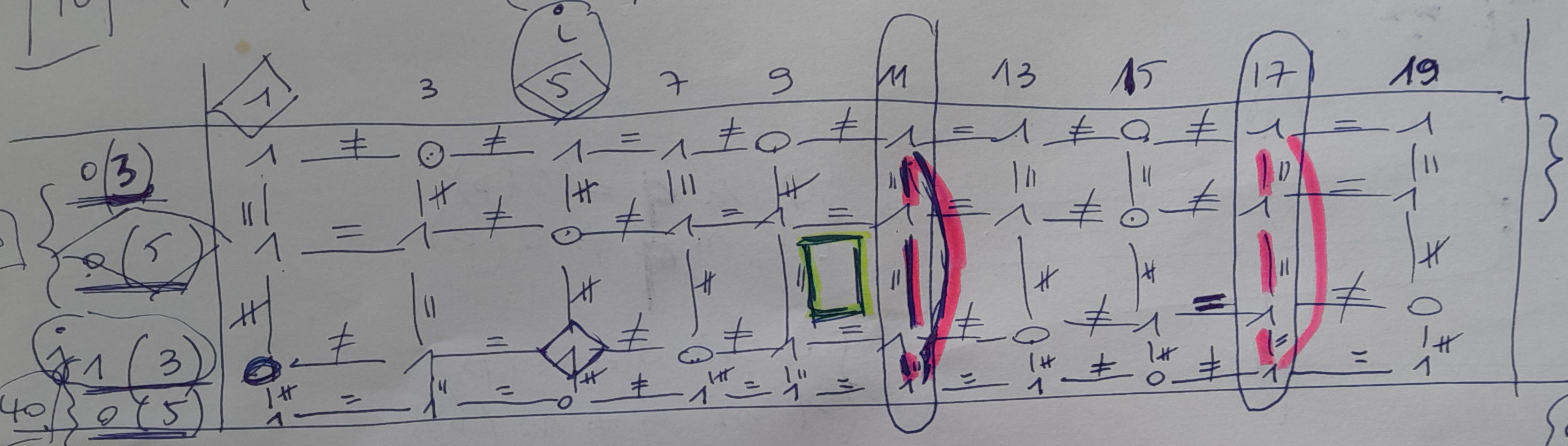
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
.0, 11.0, 11.0, 11.0, 12.0, 12.0, 12.0, 12.0, 13.0, 13.0, 14.0, 14.0, 14.0, 14.0, 15.0, 15.0, 15.0, 15.0, 15.0, 15.0, 16.0, 16.0, 16.0, 16.0, 16.0, 16.0, 17.0, 17.0, 18.0, 18.0, 18.0,
18.0, 18.0, 18.0, 19.0, 19.0, 19.0, 19.0, 20.0, 20.0, 21.0, 21.0, 21.0, 21.0, 21.0, 21.0, 22.0, 22.0, 22.0, 22.0, 23.0, 23.0, 23.0, 23.0, 23.0, 23.0, 24.0, 24.0, 24.0, 24.0, 24.0, 24.0
, 24.0, 24.0, 25.0, 25.0, 25.0, 25.0] [1.0, 2.0, 2.0, 3.0, 3.0, 4.0, 4.0, 4.0, 4.0, 5.0, 5.0, 6.0, 6.0, 6.0, 6.0, 7.0, 7.0, 8.0, 8.0, 8.0, 8.0, 9.0, 9.0, 9.0, 9.0, 9.0, 9.0, 10.0, 10.0
, 11.0, 11.0, 11.0, 11.0, 11.0, 11.0, 12.0, 12.0, 12.0, 12.0, 13.0, 13.0, 14.0, 14.0, 14.0, 14.0, 15.0, 15.0, 15.0, 15.0, 15.0, 15.0, 16.0, 16.0, 16.0, 16.0, 16.0, 16.0, 17.0, 17.0, 18
.0, 18.0, 18.0, 18.0, 18.0, 18.0, 19.0, 19.0, 19.0, 19.0, 20.0, 20.0, 21.0, 21.0, 21.0, 21.0, 21.0, 21.0, 22.0, 22.0, 22.0, 22.0, 23.0, 23.0, 23.0, 23.0, 23.0, 23.0, 24.0, 24.0, 24.0,
24.0, 24.0, 24.0, 24.0, 24.0, 25.0, 25.0, 25.0, 25.0]
```

$\boxed{40}$ 1(3) 0(5)

$2(\pi\sqrt{n}-1)$ nb $= 2(\pi(\sqrt{n})-1)-1$ écarts verticaux

$\left\lfloor \frac{n-2}{4} \right\rfloor$ écarts horizontaux



$2\left\lfloor \frac{n}{p} \right\rfloor$ signes ≠

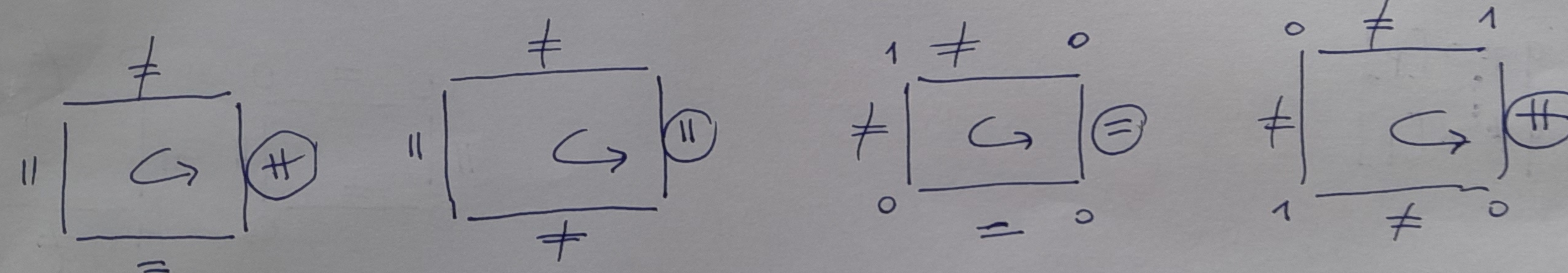
40 → 9

42 → 10

44 → 10

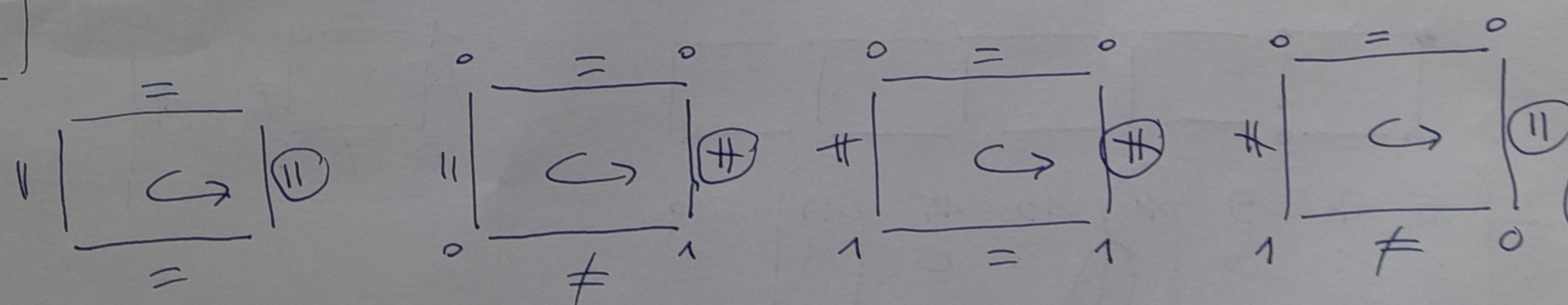
46 → 11

$$2\sum_{\substack{p<\sqrt{n}\\ \text{impair}}} \frac{n}{4p} < \left\lfloor \frac{n-2}{4} \right\rfloor + 1$$



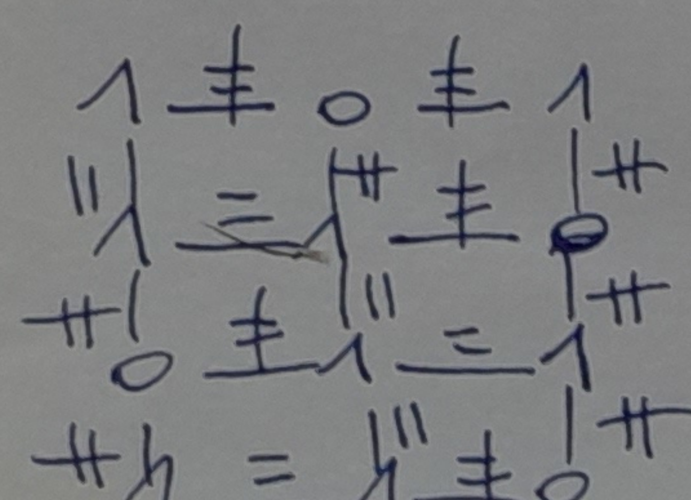
le nb de ≠ est tjs pair dans un rectangles simples.

$\boxed{\text{jade 1}}$



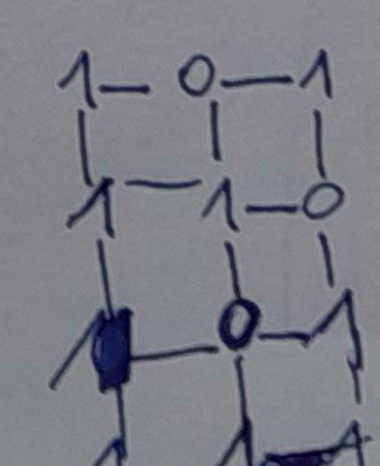
(nb impair) de ?

$\boxed{10}$



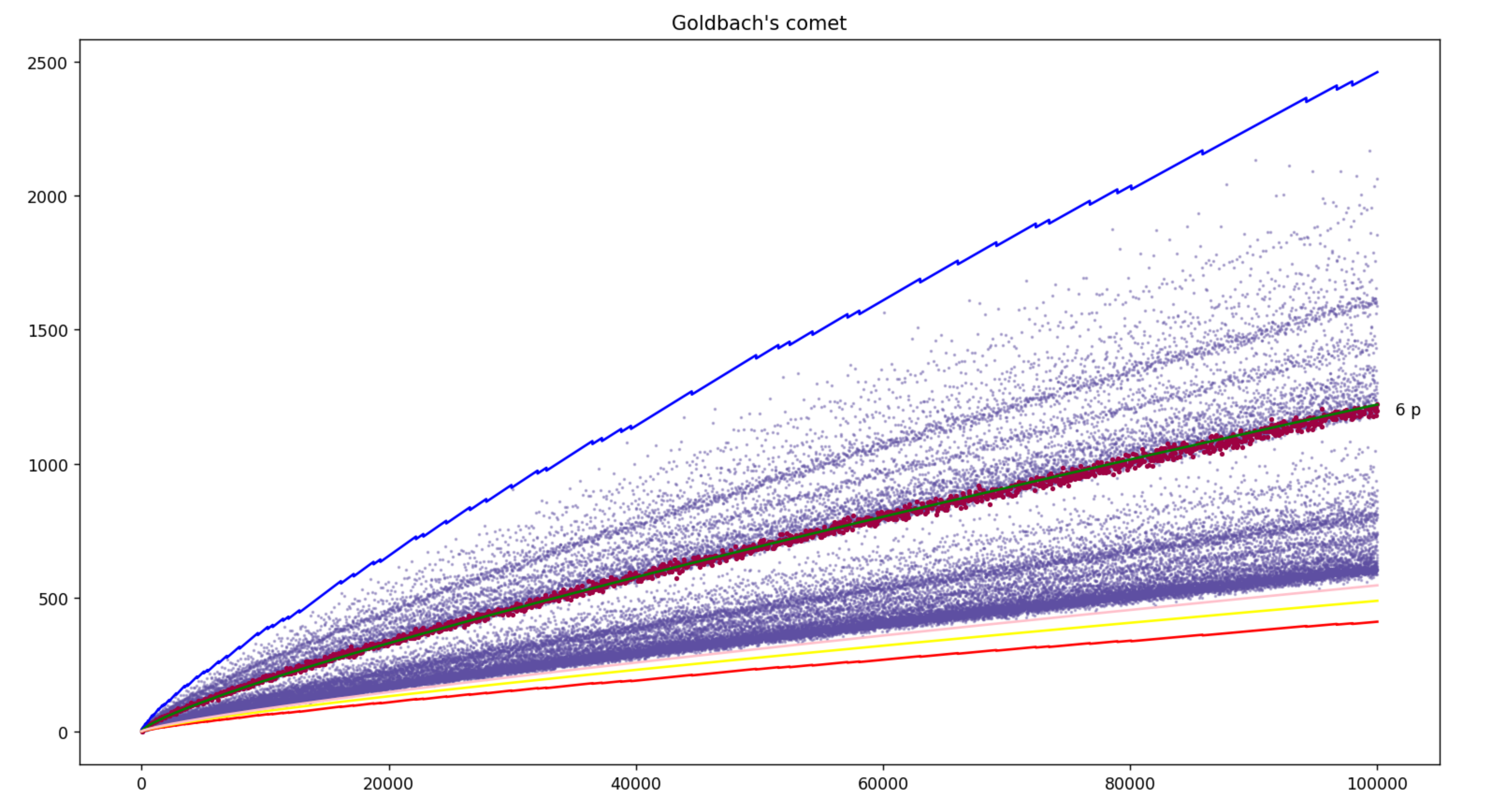
11 ≠ / 17

$\boxed{12}$



10 ≠ / 17

CG = ∃ un cycle vertical étiqueté uniquement par des = et de lgth $2(\pi\sqrt{n}-1)$



Goldbach's comet
2500
2000
1500
1000
500
0
0
20000
40000
60000
80000
100000
6 p

```python
lesx = []
lesy = []
lesymin = []
lesymax = []
for x in range(2,m+1):
    prod = 1.0
    rac = int(sqrt(x))
    for p in range(3,rac+1):
        if p in primes:
            prod = prod*(1-(2/p))
    lesx.append(x)
    lesymin.append(prod*x)
    lesymax.append(prod*x/6)
plt.plot(lesx,lesymin,color='blue')
plt.plot(lesx,lesymax,color='red')
#plt.show()
```

```
lesx = []
lesy=[]
lesy2=[]
lesy3=[]
for x in range(6,m):
    lesx.append(x)
    lesy.append((1.618033*x)/(log(x)*log(x)))
    lesy2.append((1.618033*2*x)/(5*log(x)*log(x)))
    lesy3.append((1.618033*x)/(sqrt(5)*log(x)*log(x)))
plt.plot(lesx,lesy,color='green')
plt.plot(lesx,lesy2,color='yellow')
plt.plot(lesx,lesy3,color='pink')
plt.show()
```