Une fonction de comptage des nombres premiers (Denise Vella-Chemla, 14.9.2018)

On définit ci-dessous une fonction $f_b(x)$ paramétrée par un entier b (la base, qui n'est pas forcément un nombre premier).

Cette fonction est définie sur l'ensemble des entiers naturels $\mathbb{N}$.

Elle permet d'"isoler" une puissance de la base dans l'écriture d'un nombre entier sous la forme d'un produit ainsi * :

$$f_b(kb^y) = y \text{ avec } b \not| k.$$

Cette fonction coïncide avec la valuation p-adique $v_p(n)$ qui est définie sur $\mathbb{Q}$ et selon p un nombre premier †.

$$y \text{ est l'exposant maximum tel que } b^y \mid n.$$

Les images des 30 premiers entiers par les fonctions f_2, f_3, f_4, f_5 sont :

x	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
$f_2(x)$	0	1	0	2	0	1	0	3	0	1	0	2	0	1	0	4	0	1	0	2	0	1	0	3	0	1	0	2	0	1
$f_3(x)$	0	0	1	0	0	1	0	0	2	0	0	1	0	0	1	0	0	2	0	0	1	0	0	1	0	0	3	0	0	1
$f_4(x)$	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	2	0	0	0	1	0	0	0	1	0	0	0	1	0	0
$f_5(x)$	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	2	0	0	0	0	1

Ces fonctions ont un comportement similaire à celui du logarithme : par exemple, $f_k(kx) = f_k(x) + 1$. Cependant, elles ne coïncident avec les logarithmes en base k que pour les x qui sont des puissances des différents k. Entre ces puissances, les logarithmes en base k croissent lentement sur $\mathbb{R}$ tandis que les $f_k(x)$ sont très souvent nulles et entrecoupées de valeurs erratiques tous les k nombres.

On remarque dans chaque séquence les multiples occurrences d'une sous-séquence répétitive, qui rappelle la notion de motif musical, ces motifs étant palindromiques ‡ :

- multiples occurrences du "motif" 010 dans la séquence des $f_2(x)$, entre lesquelles s'intercalent les valeurs 2, 3, 2, 4, etc.
- multiples occurrences du "motif" 00100100 dans la séquence des $f_3(x)$, entre lesquelles s'intercalent les valeurs 2, 2, 3, 2, 2, 3, 2, 2, 4, etc.
- multiples occurrences du "motif" $(0^{p-1}1)^{p-1}0^{p-1}$ dans la séquence des $f_p(x)$, entre lesquelles s'intercalent les valeurs 2 ($p-1$ fois), etc.

La fonction $Sf(n,x)$ qui associe à n la somme des valeurs des $f(n,x)$ pour x un entier compris entre 2 et $\lfloor\sqrt{n}\rfloor$, associe comme image 1 aux nombres premiers (> 1) et associe comme image un entier strictement supérieur à 1 aux nombres composés.

$$Sf(n) = \sum_{2 \leq x \leq \lfloor\sqrt{n}\rfloor} f(n,x)$$

Les valeurs $Sf(x)$ sont fournies ci-dessous pour les premiers entiers :

x	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
$Sf(x)$	1	1	3	1	2	1	4	3	2	1	4	1	2	2	7	1	4	1	4	2	2	1	6	3	2	4	4	1	4

Maintenant qu'on a trouvé une fonction qui vaut 1 pour les nombres premiers et est strictement supérieure à 1 pour les nombres composés, on définit une fonction toute simple basée sur elle, et qui compte les nombres premiers inférieurs ou égaux à un entier donné :

$$\pi_D(x) = \left\lfloor \sum \frac{1}{Sf(x)} \right\rfloor$$

Cette fonction compte le nombre de nombres premiers inférieurs ou égaux à x.

∗. La présente note reprend les idées d'une note de février 2006, Fractales, symétrie et conjecture de Goldbach, http://denisevellachemla.eu/fevrier2006.pdf.

†. Les fonctions définies ici ne doivent pas être confondues avec celles fournissant pour un entier les coefficients intervenant dans leur écriture en base p, i.e. leur écriture comme combinaison linéaire de puissances de p.

‡. Les fonctions discrètes proposées ici rappellent par leur forme les spectres de fréquences de *fonctions sinusoïdales modulées en amplitude. Cf. dessin en annexe et article wikipedia sur la Modulation d'amplitude https://fr.wikipedia.org/wiki/Modulation'damplitude*

Annexe : programme de calcul de $\pi_D(x)$

```
1
2  #include <iostream>
3  #include <stdio.h>
4  #include <cmath>
5  #include <math.h>
6
7  int prime(int atester) {
8    bool pastrouve=true;
9    unsigned long k = 2;
10
11   if (atester == 1) return 0;
12   if (atester == 2) return 1;
13   if (atester == 3) return 1;
14   if (atester == 5) return 1;
15   if (atester == 7) return 1;
16   while (pastrouve)
17     if ((k * k) > atester)
18       return 1;
19     else
20       if ((atester % k) == 0) return 0 ;
21       else k++;
22 }
23
24 int main (int argc, char* argv[]) {
25   int n, nmax, p, puiss, tempo, somme ;
26   float compte ;
27   int valpadique[1000002] ;
28
29   compte = 0.0 ;
30   nmax = 1000 ;
31   for (n = 2 ; n <= nmax ; ++n) {
32       std::cout << "\n" << n << " -> " ;
33       for (p = 2 ; p <= sqrt(n) ; ++p)
34         valpadique[p] = 0 ;
35       somme = 1 ;
36       for (p = 2 ; p <= sqrt(n) ; ++p) {
37         puiss = 1 ;
38         tempo = n ;
39         while ((tempo/p > 0) && ((tempo % p) == 0)) {
40           tempo = tempo/p ;
41           puiss = puiss+1 ;
42         }
43       valpadique[p] = puiss-1 ;
44       somme = somme+valpadique[p] ;
45     }
46     compte = compte+floor(1.0/(float)somme) ;
47     if (prime(n))
48       std::cout << " premier" ;
49     else
50       std::cout << " compose" ;
51     std::cout << "   somme " << somme ;
52     std::cout << "   compte " << compte << "\n" ;
53   }
54 }
```