# Programme de la somme des diviseurs d'Euler

Denise Vella-Chemla

26/3/14

L'article d'Euler *Découverte d'une loi tout extraordinaire des nombres par rapport à la somme de leurs diviseurs* est magique. On reste subjugué par la manière dont le mathématicien a trouvé la formule récurrente de la somme des diviseurs. Même le fait de la programmer la laisse hermétique. On trouve particulièrement esthétique la manière dont les nombres pentagonaux surgissent de la combinaison par différence de la suite des entiers et de la suite des impairs.

```
#include <iostream>
#include <cmath>

const int taille=100;
int a[taille];
int h[taille];
int euler[taille];

int f(int x) { return (3 * x * x - x) / 2; }

int g(int x) { return (3 * x * x + x) / 2; }

int remplis_h()
{int i,y;

  for (i=1; i<=taille; i++)
    if (i % 2 == 0) h[i]=f(i/2);
    else h[i]=g((i-1)/2);
}

int remplis_a()
{int i;

  for (i=1; i<=taille; i++)
    if ((i % 4 == 1) || (i % 4 == 2)) a[i]=1;
    else a[i]=-1;
}
```

```
int calcule_euler()
{int x, y, somme;

  euler[1]=1;
  for (x=2; x<=taille; x++)
    {
      somme = 0; y=0;
      while (x-h[y] >= 0)
        {
    if (x == h[y]) somme = somme + a[y-1] * x;
    else somme = somme + a[y-1] * euler[x-h[y]];
    y++;
  }
      euler[x]=somme;
    }
}

int main (int argc, char* argv[])
{
  int i;

  remplis_a();
  remplis_h();
  calcule_euler();
  for (i=1 ; i <= taille ; i++)
    std::cout << i << " : " << euler[i] << "\n" ;
}
```

On peut voir les nombres premiers comme des minima locaux de la fonction somme des diviseurs.

Puisque la somme des diviseurs d'un nombre premier $p$ vaut $p+1$, $p+(n-p)$ est une décomposition de Goldbach de $n$ si et seulement si $\sigma(p)+\sigma(n-p)=n+2$. Les décomposants de Goldbach minimisent donc la somme des sommes des diviseurs de $p$ et $n-p$.