

```

#include <iostream>
#include <cmath>
#include <stdio.h>
#include <set>
#include <algorithm>
#include <functional>
#include <climits>

int tabfacteurs[2018], tabpuiss[2018], tabexpo[2018], residfacteurs[2018] ;

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

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

int pgcd(int m, int n)
{
    while (m != 0)
    {
        int r ;

        r = n % m ;
        n = m ;
        m = r ;
    }
    return(n) ;
}

int main (int argc, char* argv[]) {
    int i, j, k, p, nbdiv, nbdivmax, nbresid, nbresidbas, nbresidhaut, tempo,
    dejavu, expo, puiss, x, y, prodpmoinsun, tmp, indic ;
    float residfacteurs[2018] ;
    bool boolpremier ;
    int tab[2018] ;

    for (i = 1 ; i <= 2016 ; ++i)
        //i = 5 ;
        {
            std::cout << "\n\n" << i << " -> " ;
            prodpmoinsun=1 ;
            tab[i] = 1 ;
            tabfacteurs[i] = 1 ;
            tabpuiss[i] = 1 ;
            tabexpo[i] = 1 ;
            residfacteurs[i] = 1.0 ;
            tempo = i ; p = i/2 ; nbdiv = 1 ;
            if (prime(tempo))
                {

```

```

    tabfacteurs[1] = tempo ;
    tabpuiss[1] = tempo ;
    tabexpo[1] = 1 ;
}
else while ((tempo > 1) && (p > 1))
    {
        if ((prime(p)) && ((tempo%p) == 0))
            {
                tabfacteurs[nbdiv] = p ;
                nbdiv = nbdiv+1 ;
                tempo = tempo/p ;
            }
        p=p-1 ;
    }
if (not(prime(i))) nbdiv=nbdiv-1 ;
if ((nbdiv == 1) && (prime(i)))
{
    tabpuiss[1] = i ;
    tabexpo[1] = 1 ;
}
else if ((nbdiv == 1) && (not(prime(i))))
{
    tempo = tabfacteurs[1] ;
    tabpuiss[1] = i ;
    expo = 1 ;
    while (tempo < i)
        {
            tempo=tempo*tabfacteurs[1] ;
            expo = expo+1 ;
        }
    tabexpo[1] = expo ;
}
else if (nbdiv > 1)
{
    for (k = 1 ; k <= nbdiv ; ++k)
        {
            tempo = tabfacteurs[k] ;
            expo = 1 ;
            while (((i % tempo) == 0) && (tempo < i))
                {
                    tempo=tempo*tabfacteurs[k] ;
                    expo = expo+1 ;
                }
            tabpuiss[k] = tempo/tabfacteurs[k] ;
            tabexpo[k] = expo-1 ;
        }
}
for (k = 1 ; k <= nbdiv ; ++k)
{
    std::cout << tabfacteurs[k] << "^" ;
    std::cout << tabexpo[k] << "." ;
    for (tmp = 1 ; tmp <= tabexpo[k] ; ++tmp)
        prodpmoinsun = prodpmoinsun*(tabfacteurs[k]-1) ;
}

indic = 1 ;
for (tmp=2 ; tmp <= i-1 ; ++tmp) if (pgcd(i,tmp)==1) indic++ ;
std::cout << "\nphi -> " << indic ;

std::cout << "\n" ;

std::cout << "puissances -> " ;
tab[1] = 1 ;
for (j = 2 ; j <= i-1 ; ++j)

```

