

```

#include <iostream>
#include <stdio.h>

int tabfacteurs[10000], tabpuiss[10000], tabexpo[10000], tab0[10000],
tab1[10000], tab2[10000] ;

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 main (int argc, char* argv[]) {
    int n, k, p, nbdiv, tempo, expo, nbdivt1, nbdivt2, nbprime ;

    nbprime = 0 ;
    for (n = 1 ; n <= 1157 ; n=n+2)
    {
        std::cout << "\n" << n << " -> " ;
        tabfacteurs[n] = 1 ;
        tab0[n] = 1 ;
        tab1[n] = 1 ;
        tab2[n] = 1 ;
        tabpuiss[n] = 1 ;
        tabexpo[n] = 1 ;
        tempo = n ; p = n/2 ;
        nbdiv = 1 ;
        nbdivt1 = 0 ;
        nbdivt2 = 0 ;
        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(n))) nbdiv=nbdiv-1 ;
        if ((nbdiv == 1) && (prime(n)))
        {
            tabpuiss[1] = n ;
            tabexpo[1] = 1 ;
            if ((n % 4) == 1) nbdivt1 = nbdivt1+1 ;
            else if ((n % 4) == 3) nbdivt2 = nbdivt2+1 ;
        }
        else if ((nbdiv == 1) && (not(prime(n))))
        {
            tempo = tabfacteurs[1] ;
            tabpuiss[1] = n ;
            expo = 1 ;
            while (tempo < n)
            {
                tempo=tempo*tabfacteurs[1] ;
                expo = expo+1 ;
            }
            tabexpo[1] = expo ;
            if ((tabfacteurs[1] % 4) == 1) nbdivt1 = nbdivt1+expo ;
            else if ((tabfacteurs[1] % 4) == 3) nbdivt2 = nbdivt2+expo ;
        }
        else if (nbdiv > 1)
        {
            for (k = 1 ; k <= nbdiv ; ++k)
            {
                tempo = tabfacteurs[k] ;
                expo = 1 ;
                while (((n % tempo) == 0) && (tempo < n))
                {
                    tempo=tempo*tabfacteurs[k] ;
                    expo = expo+1 ;
                }
            }
        }
    }
}

```

```

        tabpuiss[k] = tempo/tabfacteurs[k] ;
        tabexpo[k] = expo-1 ;
        if ((tabfacteurs[k] % 4) == 1) nbdivt1 = nbdivt1+expo-1 ;
        else if ((tabfacteurs[k] % 4) == 3) nbdivt2 = nbdivt2+expo-1 ;
    }
}
for (k = 1 ; k <= nbdiv ; ++k)
{
    std::cout << tabfacteurs[k] << "^" ;
    std::cout << tabexpo[k] << "." ;
}
tab0[n] = nbdiv ;
tab1[n] = nbdivt1 ;
tab2[n] = nbdivt2 ;
}
std::cout << "\n\n" ;
std::cout << "Nombres de diviseurs 4k+1 \n" ;
for (n = 1 ; n <= 1157 ; n=n+2)
    std::cout << tab1[n] << " " ;
std::cout << "\n\n" ;
std::cout << "Nombres de diviseurs 4k+3 \n" ;
for (n = 1 ; n <= 1157 ; n=n+2)
    std::cout << tab2[n] << " " ;
std::cout << "\n\n" ;
std::cout << "Nombres total de facteurs premiers différents\n" ;
for (n = 1 ; n <= 1157 ; n=n+2)
    std::cout << tab0[n] << " " ;
}

```