

```

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

int prime(int atester)
{
    unsigned long diviseur=2;
    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, x, xa, xb, xc, xd, xaprim, xbprim, xcprim, xdprim ;
    int pix, pi2x, compteprem, debuthaut, za, zc, ya, yc ;

    for (n=14 ; n <= 10000 ; n=n+2)
    {
        xa=0 ; xb=0 ; xc=0 ; xd=0 ; ya=0 ; yc=0; za=0 ; zc=0 ;
        for (x = 6 ; x <= (n+4)/2 ; x=x+2)
        {
            if (prime(x-3)) za++ ;
            else zc++ ;
        }
        if ((n/2) % 2 == 0) debuthaut = ((n+4)/2)+2 ;
        else debuthaut = ((n+4)/2)+1 ;
        for (x = debuthaut ; x <= n ; x=x+2)
        {
            if (prime(x-3)) ya++ ;
            else yc++ ;
        }
        for (x = 3 ; x <= n/2 ; x=x+2)
        {
            if (prime(x)) { if (prime(n-x)) xa++ ; else xc++ ;}
            else { if (prime(n-x)) xb++ ; else xd++ ; }
        }
        pix=0 ; pi2x = 0 ;
        for (compteprem=2 ; compteprem <= n ; compteprem++)
        {
            if (prime(compteprem)) {
                if (compteprem <= n/2) pix++ ;
                pi2x++ ;
            }
        }
        std::cout << "\n\n" << n << " : \n" ;
        //std::cout << "(n-2)/8 " << (n-2)/8 ;
        std::cout << "n/4 " << n/4 << " " ;
        std::cout << " pi(n) " << pi2x << " " ;
        std::cout << " pi(n/2) " << pix << "\n";
        printf("Xa %2d ",xa) ;
    }
}

```

```

printf("Xb %2d ",xb) ;
printf("Xc %2d ",xc) ;
printf("Xd %2d ",xd) ;
std::cout << "\n" ;
printf("Ya %2d ",ya) ;
printf("Yc %2d ",yc) ;
std::cout << "\n" ;
printf("Za %2d ",za) ;
printf("Zc %2d ",zc) ;
//if (yc > (n-2)/8) std::cout << "Yc a dépassé la moitié du haut !\n" ;
//else std::cout << "Yc plus petit que la moitié du haut !\n" ;
//if (zc > (n-2)/8) std::cout << "Zc a dépassé la moitié du bas !\n" ;
//else std::cout << "Zc plus petit que la moitié du bas !\n" ;
//if ((yc>0) &&
//    (zc>0) &&
//    (xd > yc+zc-(n-4)/4) &&
//    (yc > (n-2)/8) &&
//    (zc > (n-2)/8))
//    std::cout << "Xd est devenu assez grand pour tirer Xa !\n" ;
//else
//    std::cout << "Xd n'est pas encore assez grand pour tirer Xa !\n" ;
}

```