Programme en C++ pour les sommes alternées de cosinus (les $4k+3$ ont pour image 0 et les $4k+1$ ont pour image 1)

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

using namespace std ;
typedef complex<double> dcomp ;

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

  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, i, j ;
  double somme ;
  int oppose ;
  const double PI = 4.0 * atan(1.0);

  for (n = 2 ; n <= 50 ; n++)
    {
      oppose = 1 ;
      somme = 0.0 ;
      for (i = 2 ; i <= n-1 ; i++)
  {
    std::cout << "\n" ;
    for (j = 1 ; j <= i ; j++)
      {
        oppose = (-1) * oppose ;
        somme += oppose * cos(2.0 * PI * (double) n * (double) j / (double) i) ;
        std::cout << n << "," << i << "," << j << " -> " ;
        std::cout << 360 * (double) n * (double) j / (double) i << " " ;
        std::cout << oppose * cos(2.0 * PI * (double) n * (double) j / (double) i) << "\n" ;
      }
  }
      somme = somme-1-oppose*0.5 ;
      std::cout << n << " somme globale " << somme << "\n" ;
    }
}
```

```
3,2,1 -> 540  1
3,2,2 -> 1080  1
3 somme globale 0.5

4,2,1 -> 720  -1
4,2,2 -> 1440  1

4,3,1 -> 480  0.5
4,3,2 -> 960  -0.5
4,3,3 -> 1440  -1
4 somme globale -1.5

5,2,1 -> 900  1
5,2,2 -> 1800  1

5,3,1 -> 600  0.5
5,3,2 -> 1200  -0.5
5,3,3 -> 1800  -1

5,4,1 -> 450  3.06162e-16
5,4,2 -> 900  1
5,4,3 -> 1350  -2.69484e-15
5,4,4 -> 1800  -1
5 somme globale 0.5

6,2,1 -> 1080  -1
6,2,2 -> 2160  1

6,3,1 -> 720  -1
6,3,2 -> 1440  1
6,3,3 -> 2160  -1

6,4,1 -> 540  -1
6,4,2 -> 1080  -1
6,4,3 -> 1620  -1
6,4,4 -> 2160  -1

6,5,1 -> 432  0.309017
6,5,2 -> 864  0.809017
6,5,3 -> 1296  -0.809017
6,5,4 -> 1728  -0.309017
6,5,5 -> 2160  1
6 somme globale -5.5

7,2,1 -> 1260  1
7,2,2 -> 2520  1

7,3,1 -> 840  0.5
7,3,2 -> 1680  -0.5
7,3,3 -> 2520  -1

7,4,1 -> 630  -4.28626e-16
7,4,2 -> 1260  1
7,4,3 -> 1890  -4.90478e-16
7,4,4 -> 2520  -1

7,5,1 -> 504  -0.809017
7,5,2 -> 1008  -0.309017
7,5,3 -> 1512  0.309017
7,5,4 -> 2016  0.809017
7,5,5 -> 2520  1
```

```
7,6,1 -> 420  -0.5
7,6,2 -> 840  -0.5
7,6,3 -> 1260  1
7,6,4 -> 1680  -0.5
7,6,5 -> 2100  -0.5
7,6,6 -> 2520  1
7 somme globale 0.5

8,2,1 -> 1440  -1
8,2,2 -> 2880  1

8,3,1 -> 960  0.5
8,3,2 -> 1920  -0.5
8,3,3 -> 2880  -1

8,4,1 -> 720  1
8,4,2 -> 1440  -1
8,4,3 -> 2160  1
8,4,4 -> 2880  -1

8,5,1 -> 576  -0.809017
8,5,2 -> 1152  -0.309017
8,5,3 -> 1728  0.309017
8,5,4 -> 2304  0.809017
8,5,5 -> 2880  1

8,6,1 -> 480  0.5
8,6,2 -> 960  -0.5
8,6,3 -> 1440  -1
8,6,4 -> 1920  -0.5
8,6,5 -> 2400  0.5
8,6,6 -> 2880  1

8,7,1 -> 411.429  -0.62349
8,7,2 -> 822.857  -0.222521
8,7,3 -> 1234.29  0.900969
8,7,4 -> 1645.71  -0.900969
8,7,5 -> 2057.14  0.222521
8,7,6 -> 2468.57  0.62349
8,7,7 -> 2880  -1
8 somme globale -1.5

9,2,1 -> 1620  1
9,2,2 -> 3240  1

9,3,1 -> 1080  -1
9,3,2 -> 2160  1
9,3,3 -> 3240  -1

9,4,1 -> 810  5.51091e-16
9,4,2 -> 1620  1
9,4,3 -> 2430  -3.42963e-15
9,4,4 -> 3240  -1

9,5,1 -> 648  0.309017
9,5,2 -> 1296  0.809017
9,5,3 -> 1944  -0.809017
9,5,4 -> 2592  -0.309017
9,5,5 -> 3240  1
```

```
9,6,1 -> 540  1
9,6,2 -> 1080  1
9,6,3 -> 1620  1
9,6,4 -> 2160  1
9,6,5 -> 2700  1
9,6,6 -> 3240  1

9,7,1 -> 462.857  0.222521
9,7,2 -> 925.714  -0.900969
9,7,3 -> 1388.57  -0.62349
9,7,4 -> 1851.43  0.62349
9,7,5 -> 2314.29  0.900969
9,7,6 -> 2777.14  -0.222521
9,7,7 -> 3240  -1

9,8,1 -> 405  0.707107
9,8,2 -> 810  -5.51091e-16
9,8,3 -> 1215  -0.707107
9,8,4 -> 1620  1
9,8,5 -> 2025  -0.707107
9,8,6 -> 2430  3.42963e-15
9,8,7 -> 2835  0.707107
9,8,8 -> 3240  -1
9 somme globale 6.5

10,2,1 -> 1800  -1
10,2,2 -> 3600  1

10,3,1 -> 1200  0.5
10,3,2 -> 2400  -0.5
10,3,3 -> 3600  -1

10,4,1 -> 900  -1
10,4,2 -> 1800  -1
10,4,3 -> 2700  -1
10,4,4 -> 3600  -1

10,5,1 -> 720  1
10,5,2 -> 1440  -1
10,5,3 -> 2160  1
10,5,4 -> 2880  -1
10,5,5 -> 3600  1


10,6,1 -> 600  0.5
10,6,2 -> 1200  -0.5
10,6,3 -> 1800  -1
10,6,4 -> 2400  -0.5
10,6,5 -> 3000  0.5
10,6,6 -> 3600  1

10,7,1 -> 514.286  0.900969
10,7,2 -> 1028.57  0.62349
10,7,3 -> 1542.86  0.222521
10,7,4 -> 2057.14  -0.222521
10,7,5 -> 2571.43  -0.62349
10,7,6 -> 3085.71  -0.900969
10,7,7 -> 3600  -1
```

```
10,8,1 -> 450  3.06162e-16
10,8,2 -> 900  1
10,8,3 -> 1350  -2.69484e-15
10,8,4 -> 1800  -1
10,8,5 -> 2250  -2.45548e-16
10,8,6 -> 2700  1
10,8,7 -> 3150  -3.91949e-15
10,8,8 -> 3600  -1

10,9,1 -> 400  0.766044
10,9,2 -> 800  -0.173648
10,9,3 -> 1200  -0.5
10,9,4 -> 1600  0.939693
10,9,5 -> 2000  -0.939693
10,9,6 -> 2400  0.5
10,9,7 -> 2800  0.173648
10,9,8 -> 3200  -0.766044
10,9,9 -> 3600  1
10 somme globale -5.5
```