

```

1 /* slowFT-linear-asIs.c: Fourier transform "as-is" by S. Tanaka, 2006 */
2 #include <math.h>
3 #include <stdio.h>
4 #include <time.h>
5
6 #define twoPi 3.1415926535 * 2.0
7 #define N 512
8
9 FILE *fpW;
10 FILE *fopen();
11
12 void main( void )
13 {
14     int t, fn;
15     double dataR[N], dataI[N];
16     double wR[N][N], wI[N][N], DataR[N], DataI[N];
17     double ffR, ffI, power;
18
19     double start, finish;
20
21     start=clock();
22     for (t=0; t<N; t++) {
23         dataR[t] = 1.0* cos(2*twoPi*t/N) /* データの生成 */
24             +1.0* cos(4*twoPi*t/N);
25         dataI[t] = 0.0;
26     }
27     for (fn=0; fn<N; fn++) {
28         for (t=0; t<N; t++) {
29             wR[t][fn] = cos(fn*twoPi*t/N);
30             wI[t][fn] = -sin(fn*twoPi*t/N);
31         }
32     }
33     ffR = 0.0;
34     ffI = 0.0;
35     for (fn=0; fn<N; fn++) {
36         for (t=0; t<N; t++) {
37             ffR = ffR + dataR[t]*wR[t][fn] - dataI[t]*wI[t][fn];
38             ffI = ffI + dataR[t]*wI[t][fn] + dataI[t]*wR[t][fn];
39         }
40         DataR[fn] = ffR;
41         DataI[fn] = ffI;
42         ffR = 0.0;
43         ffI = 0.0;
44     }
45     finish=clock();
46
47     fpW = fopen("slowFT-linear-asIs.txt", "w");
48     fprintf(fpW, "          離散フーリエ変換¥n");
49     fprintf(fpW, "t, fn    f(t)          Re(F(fn))  Im(F(fn))  振幅¥n");
50
51     for (fn=0; fn<N; fn++) {
52         power = DataR[fn]*DataR[fn] + DataI[fn]*DataI[fn];
53         fprintf(fpW, " %3d %10.6f %10.6f %10.6f %10.6f ¥n",
54             fn, dataR[fn], DataR[fn], DataI[fn], power);
55     }
56     printf("Elapse time: %lf seconds!¥n", (double)(finish-start)/CLOCKS_PER_SEC);
57     printf("Result written to 'slowFT-linear-asIs.txt.'¥n");
58
59 }
60

```