toto-share.com

sharing everything

WEB HOSTING MONEY ONLINE ASCII TABLE ABOUT ME

You are here: Home / 2012 / July / *C/C++ : fftw Tutorial*

C/C++ : fftw Tutorial

By totosugito on July 10, 2012

C/C++ : fftw Tutorial

FFTW is a C subroutine library for computing the discrete Fourier transform (DFT) in one or more dimensions, of arbitrary input size, and of both real and complex data. I use this library for compute FFT because the library is fast and simple to use. This is my **fftw tutorial**. So, from this **fftw tutorial**, you can know how to use **fftw library** to process

Search

Categories

- .NET (3)
- Android (1)
- C (37)
- C++/Qt (41)
- freelance (6)
- Internet (18)
- Linux (49)
- Matlab (9)
- Notes (4)
- Python (7)
- Windows (10)

your data.

I create four function in this **fftw tutorial**. The function is **how to compute fft and ifft** for your data with input/output in floating point and double data format. You can use this **fftw tutorial** function like at Matlab function. When using fft and ifft in Matlab, we can use this command :

```
B = fft(A)
B = fft(A, nfft)
C = ifft(B)
C = ifft(B, nfft)
```

You can use my **fftw tutorial** function like the Matlab parameter :

```
B = fftwf_data(A, lenA, nfft);
C = ifftwf_data(B, lenC, nfft);
```

The fftw accepted input data with **fftwf_complex** (floating point data format) and **fftw_complex** (double data format). This data format (**fftwf_complex** and **fftw_complex**) is an array 2D with length of colum = 2. The first column is real data and the second column is imaginer data.

This is the sample main program for this **fftw tutorial** for more explanation how to use **fftw library** in our program (**fftw sample code**).

```
/*
 * main.c
 *
 *  Created on: Jul 10, 2012
 *      Author: toto
 */

```

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
#include <time.h>
#include <fftw3.h>

int main( int argc, char** argv )
{
    fftwf_complex *data;
    fftwf_complex *fdata;
    fftwf_complex *idata;
    int ndata, nfft, i;
    unsigned int seed = 1234;

    if(argc<2)
        return(1);
    sscanf(argv[1], "%i", &ndata);
    sscanf(argv[2], "%i", &nfft);

    //allocate memory for data
    data = ( fftwf_complex* ) fftwf_malloc( sizeof( ff

    // CREATE INPUT DATA
    srand ( seed );
    for( i=0; i<ndata; i++ ) {
        data[i][0] = (double) (rand()/12345678);    //
        data[i][1] = 0.0;       //imaginer data
    }
    //print input data
    printf("Input Data \n");
    for( i = 0 ; i < ndata ; i++ )
        printf( "%i.  %2.5f  %2.5f \n", i, data[i][0],

    // FFT PROCESS
    fdata = fftwf_data(data, ndata, nfft);
    printf("\nFFT Result \n");
    for( i = 0 ; i < nfft ; i++ )
        printf( "%i.  %2.5f  %2.5f \n", i, fdata[i][0]

    // INVERS FFT PROCESS
    idata = ifftwf_data(fdata, ndata, nfft);
    printf("\nIFFT Result \n");
    for( i = 0 ; i < ndata ; i++ )
        printf( "%i.  %2.5f  %2.5f \n", i, idata[i][0]
    printf("\n");

    fftwf_free( data );
    fftwf_free( fdata );
    fftwf_free( idata );

    return 0;
}
```

This is the sample output from this **fftw tutorial code** :

```
toto@toto-laptop:~/Desktop$ fftw_demo 5 10
Input Data
0.  38.00000  0.00000
1.  37.00000  0.00000
2.  77.00000  0.00000
3.  85.00000  0.00000
4.  99.00000  0.00000

FFT Result
0.  336.00000  0.00000
1.  -14.63119  -234.00996
2.  -51.03444  63.66779
3.  63.63119  -34.12248
4.  -21.96556  28.83423
5.  92.00000  0.00000
6.  -21.96556  -28.83423
7.  63.63119  34.12248
8.  -51.03444  -63.66779
9.  -14.63119  234.00996

IFFT Result
```

```
0.   38.00000  0.00000
1.   37.00000  0.00000
2.   77.00001  0.00000
3.   85.00000  0.00000
4.   99.00000  0.00000
```

If you compare the value from this **fftw tutorial**, you will get the same value from Matlab. You can donwload the complete code from this **fftw tutorial** at here.

Source :

http://www.fftw.org/

totosugito

Hi, This is my notes when I try something. Maybe, this is NOT THE BEST SOLUTION for your problems. If you have other method or idea that better with my post, please share in this blog. Thank for visiting my site.

More Posts - Website

Related posts:

1. Qt QListView Tutorial
2. C/C++ : LibTiff Tutorial
3. Qt: QTableView Tutorial I
4. C/C++ Read Binary File
5. C/C++ create random noise (gaussian noise/white noise)

Posted in C, C++/Qt | Tagged compare fftw with matlab, fft like matlab, fftw sample code, fftw tutorial code, how to compute fft and ifft

One Response

rafael

May 16, 2013 at 5:02 pm | Permalink | **Reply**

Boa Tarde Toto.

o programa não funcionou. qual é a utilidade do arquivo Makefile?

Leave a Reply

Name *

Email *

Website

Comment

Type the text

Privacy & Terms

Post Comment

← **Previous**

Next →

Powered by **WordPress** and **Retro-fitted**

69751