

Proyecto CPP2020

Generated by Doxygen 1.8.13

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	3
2.1	File List	3
3	Class Documentation	5
3.1	Complex Class Reference	5
3.1.1	Detailed Description	5
3.2	Datos Class Reference	5
3.2.1	Detailed Description	6
3.2.2	Member Function Documentation	6
3.2.2.1	bitReversef()	6
3.2.2.2	funcion()	6
3.2.2.3	obtener_datos()	7
3.3	DFT_1d Class Reference	7
3.3.1	Detailed Description	7
3.3.2	Member Function Documentation	7
3.3.2.1	DFT()	8
3.4	FFT_1d Class Reference	9
3.4.1	Detailed Description	9
3.4.2	Member Function Documentation	9
3.4.2.1	FFT()	9

4	File Documentation	11
4.1	/home/daniel/Escritorio/Proyecto/src/apps/DFT.cpp File Reference	11
4.1.1	Detailed Description	11
4.2	/home/daniel/Escritorio/Proyecto/src/apps/FFT.cpp File Reference	12
4.2.1	Detailed Description	13
4.3	/home/daniel/Escritorio/Proyecto/src/lib/include/Complex.hpp File Reference	13
4.3.1	Detailed Description	14
4.4	/home/daniel/Escritorio/Proyecto/src/lib/include/Datos.hpp File Reference	14
4.4.1	Detailed Description	15
4.5	/home/daniel/Escritorio/Proyecto/src/lib/include/DFT_1d.hpp File Reference	16
4.5.1	Detailed Description	16
4.6	/home/daniel/Escritorio/Proyecto/src/lib/include/FFT_1d.hpp File Reference	17
4.6.1	Detailed Description	18
	Index	19

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Complex	Class Complex	5
Datos	Class Datos	5
DFT_1d	Class DFT_1d	7
FFT_1d	Class FFT_1d	9

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

/home/daniel/Escritorio/Proyecto/src/apps/ DFT.cpp	
Algoritmo para obtener el tiempo de calculo de DFT	11
/home/daniel/Escritorio/Proyecto/src/apps/ FFT.cpp	
Algoritmo para obtener el tiempo de calculo de FFT	12
/home/daniel/Escritorio/Proyecto/src/lib/include/ Complex.hpp	
Clase para almacenar datos tipo complejos	13
/home/daniel/Escritorio/Proyecto/src/lib/include/ Datos.hpp	
Clase para obtener y manipular los datos de entrada para los algoritmos	14
/home/daniel/Escritorio/Proyecto/src/lib/include/ DFT_1d.hpp	
Clase para calcular la DFT	16
/home/daniel/Escritorio/Proyecto/src/lib/include/ FFT_1d.hpp	
Clase para calcular la FFT	17

Chapter 3

Class Documentation

3.1 Complex Class Reference

Clase [Complex](#).

```
#include <Complex.hpp>
```

Public Attributes

- double [Re](#) = 0
Parte real.
- double [Im](#) = 0
Parte imaginaria.

3.1.1 Detailed Description

Clase [Complex](#).

Esta clase contiene dos variables tipo double para representar un numero complejo.

The documentation for this class was generated from the following file:

- /home/daniel/Escritorio/Proyecto/src/lib/include/[Complex.hpp](#)

3.2 Datos Class Reference

Clase [Datos](#).

```
#include <Datos.hpp>
```

Public Member Functions

- void `obtener_datos` (`Complex` *data, int N)
Metodo para obtener un array de tamaño N y guardarlo en data.
- void `bitReversef` (`Complex` *entrada, `Complex` *salida, int n, int N)
Metodo para obtener un array nuevo en orden Bit-Reversal para el calculo del FFT.
- double `funcion` (double x, int N)
Metodo para obtener el valor de una funcion.

3.2.1 Detailed Description

Clase `Datos`.

Esta clase contiene varios metodos para la obtencion y manipulacion de datos.

3.2.2 Member Function Documentation

3.2.2.1 `bitReversef()`

```
void Datos::bitReversef (
    Complex * entrada,
    Complex * salida,
    int n,
    int N )
```

Metodo para obtener un array nuevo en orden Bit-Reversal para el calculo del FFT.

Parameters

in	<i>*entrada</i>	Array en orden normal de los datos.
in	<i>*salida</i>	Array en orden Bit-Reversal de los datos.
in	<i>n</i>	Cantidad de bits de los valores del array.
in	<i>N</i>	Tamaño del array.

3.2.2.2 `funcion()`

```
double Datos::funcion (
    double x,
    int N )
```

Metodo para obtener el valor de una funcion.

Parameters

in	x	La variable independiente de la funcion.
in	N	Es el entero que se usara para el calculo de funcion.

3.2.2.3 obtener_datos()

```
void Datos::obtener_datos (
    Complex * data,
    int N )
```

Metodo para obtener un array de tamaño N y guardarlo en data.

Parameters

in	<i>*data</i>	Array donde se almacenaran los datos.
in	N	Tamaño del array.

The documentation for this class was generated from the following files:

- /home/daniel/Escritorio/Proyecto/src/lib/include/Datos.hpp
- /home/daniel/Escritorio/Proyecto/src/lib/bin/Datos.cpp

3.3 DFT_1d Class Reference

Clase [DFT_1d](#).

```
#include <DFT_1d.hpp>
```

Public Member Functions

- void [DFT](#) ([Complex](#) *array, int N)
Metodo para calcular la DFT de un array.

3.3.1 Detailed Description

Clase [DFT_1d](#).

Esta clase contiene un metodo para calcular la DFT de un array.

3.3.2 Member Function Documentation

3.3.2.1 DFT()

```
void DFT_ld::DFT (
    Complex * array,
    int N )
```

Metodo para calcular la DFT de un array.

Parameters

in	<i>*array</i>	Array de los datos de entrada para la DFT y donde se guardaran los calculos.
in	<i>N</i>	Tamaño del array.

The documentation for this class was generated from the following files:

- /home/daniel/Escritorio/Proyecto/src/lib/include/DFT_1d.hpp
- /home/daniel/Escritorio/Proyecto/src/lib/bin/DFT_1d.cpp

3.4 FFT_1d Class Reference

Clase [FFT_1d](#).

```
#include <FFT_1d.hpp>
```

Public Member Functions

- void [FFT](#) (int N, [Complex](#) *array, int E)
Metodo para calcular la FFT de un array.

3.4.1 Detailed Description

Clase [FFT_1d](#).

Esta clase contiene un metodo para calcular la FFT de un array.

3.4.2 Member Function Documentation

3.4.2.1 FFT()

```
void FFT_1d::FFT (
    int N,
    Complex * array,
    int E )
```

Metodo para calcular la FFT de un array.

Parameters

in	<i>N</i>	Tamaño del array.
in	<i>*array</i>	Array de los datos de entrada para la DFT y donde se guardaran los calculos.
in	<i>E</i>	Cantidad total de etapas.

The documentation for this class was generated from the following files:

- [/home/daniel/Escritorio/Proyecto/src/lib/include/FFT_1d.hpp](#)
- [/home/daniel/Escritorio/Proyecto/src/lib/bin/FFT_1d.cpp](#)

Chapter 4

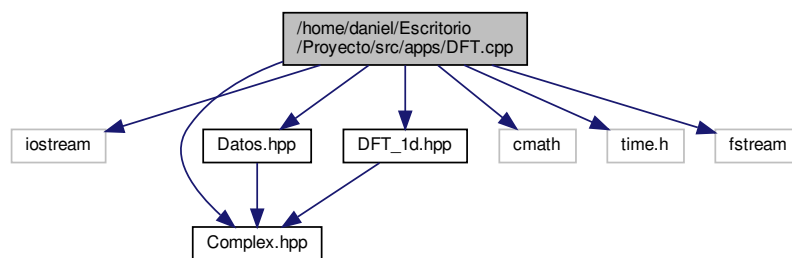
File Documentation

4.1 /home/daniel/Escritorio/Proyecto/src/apps/DFT.cpp File Reference

Algoritmo para obtener el tiempo de calculo de DFT.

```
#include <iostream>
#include <Complex.hpp>
#include <Datos.hpp>
#include <DFT_1d.hpp>
#include <cmath>
#include <time.h>
#include <fstream>
```

Include dependency graph for DFT.cpp:



Functions

- int **main** ()

4.1.1 Detailed Description

Algoritmo para obtener el tiempo de calculo de DFT.

Hallar el tiempo demorado para calcular la DFT de distintos tamaños de arrays.

Author

Daniel Reyes Barrera

Version

1.0

Date

2021

Copyright

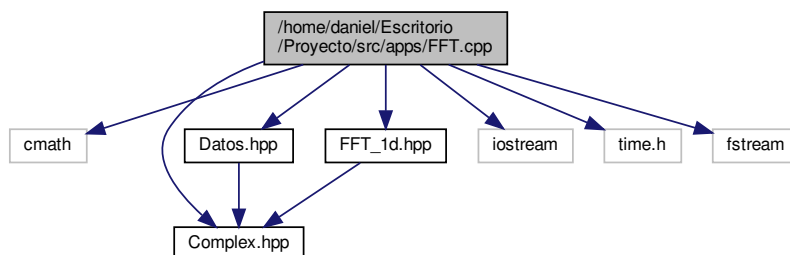
GNU Public License.

4.2 /home/daniel/Escritorio/Proyecto/src/apps/FFT.cpp File Reference

Algoritmo para obtener el tiempo de calculo de FFT.

```
#include <cmath>
#include <Complex.hpp>
#include <Datos.hpp>
#include <FFT_1d.hpp>
#include <iostream>
#include <time.h>
#include <fstream>
```

Include dependency graph for FFT.cpp:

**Functions**

- `int main ()`

4.2.1 Detailed Description

Algoritmo para obtener el tiempo de calculo de FFT.

Hallar el tiempo demorado para calcular la FFT de distintos tamaños de arrays.

Author

Daniel Reyes Barrera

Version

1.0

Date

2021

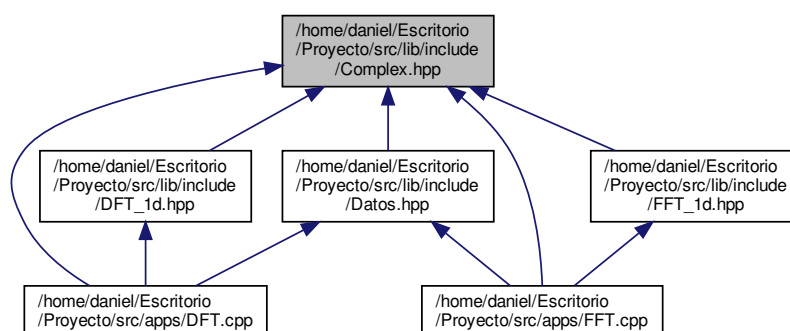
Copyright

GNU Public License.

4.3 /home/daniel/Escritorio/Proyecto/src/lib/include/Complex.hpp File Reference

Clase para almacenar datos tipo complejos.

This graph shows which files directly or indirectly include this file:



Classes

- class [Complex](#)
Clase [Complex](#).

4.3.1 Detailed Description

Clase para almacenar datos tipo complejos.

Una clase simple que contine dos variables tipo double.

Author

Daniel Reyes Barrera

Version

1.0

Date

2021

Copyright

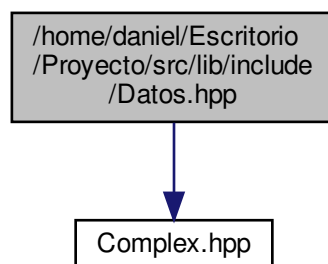
GNU Public License.

4.4 /home/daniel/Escritorio/Proyecto/src/lib/include/Datos.hpp File Reference

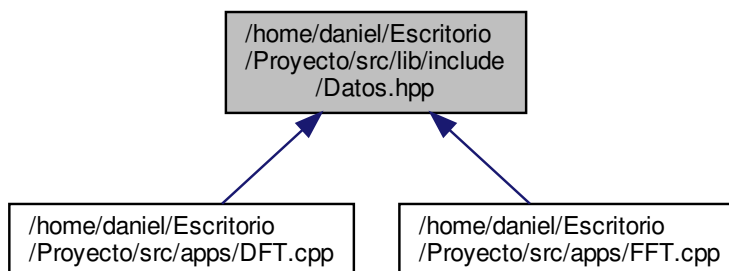
Clase para obtener y manipular los datos de entrada para los algoritmos.

```
#include <Complex.hpp>
```

Include dependency graph for Datos.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [Datos](#)
Clase [Datos](#).

4.4.1 Detailed Description

Clase para obtener y manipular los datos de entrada para los algoritmos.

Una clase que contiene varios metodos para la obtencion y manipulacion de datos.

Author

Daniel Reyes Barrera

Version

1.0

Date

2021

Copyright

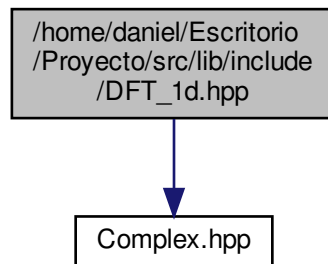
GNU Public License.

4.5 /home/daniel/Escritorio/Proyecto/src/lib/include/DFT_1d.hpp File Reference

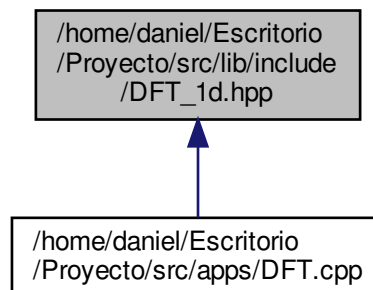
Clase para calcular la DFT.

```
#include <Complex.hpp>
```

Include dependency graph for DFT_1d.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [DFT_1d](#)
Clase [DFT_1d](#).

4.5.1 Detailed Description

Clase para calcular la DFT.

Una clase que contiene un metodo para calcular la DFT de un array.

Author

Daniel Reyes Barrera

Version

1.0

Date

2021

Copyright

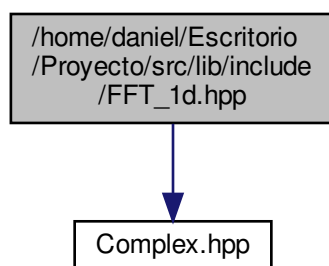
GNU Public License.

4.6 /home/daniel/Escritorio/Proyecto/src/lib/include/FFT_1d.hpp File Reference

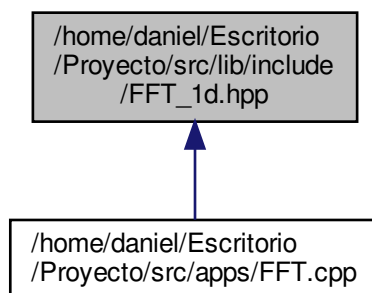
Clase para calcular la FFT.

```
#include <Complex.hpp>
```

Include dependency graph for FFT_1d.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [FFT_1d](#)
Clase [FFT_1d](#).

4.6.1 Detailed Description

Clase para calcular la FFT.

Una clase que contiene un metodo para calcular la FFT de un array.

Author

Daniel Reyes Barrera

Version

1.0

Date

2021

Copyright

GNU Public License.

Index

/home/daniel/Escritorio/Proyecto/src/apps/DFT.cpp, [11](#)
/home/daniel/Escritorio/Proyecto/src/apps/FFT.cpp, [12](#)
/home/daniel/Escritorio/Proyecto/src/lib/include/↔
 Complex.hpp, [13](#)
/home/daniel/Escritorio/Proyecto/src/lib/include/DFT_↔
 1d.hpp, [16](#)
/home/daniel/Escritorio/Proyecto/src/lib/include/↔
 Datos.hpp, [14](#)
/home/daniel/Escritorio/Proyecto/src/lib/include/FFT_↔
 1d.hpp, [17](#)

bitReversef
 Datos, [6](#)

Complex, [5](#)

DFT_1d, [7](#)
 DFT, [7](#)

DFT
 DFT_1d, [7](#)

Datos, [5](#)
 bitReversef, [6](#)
 funcion, [6](#)
 obtener_datos, [7](#)

FFT_1d, [9](#)
 FFT, [9](#)
FFT
 FFT_1d, [9](#)

funcion
 Datos, [6](#)

obtener_datos
 Datos, [7](#)