My doxygen example

Generated by Doxygen 1.8.19

Chapter 1

Main Page

This short program contains some sample code illustrating how the doxygen comments must appear

- for documenting a file, and in particular
- for documenting functions

when we intend to use the doxygen tool for preparing HTML documentation of our code. Note that this file is intended *only* to illustrate a particular set of **commenting conventions** and how they show up when implemented with doxygen. Your actual requirements may not be the same as shown here. Im particular you may need less (or more) than is shown here.

Pay careful attention, in the source code,
to the distinction between the special doxygen comments,
which produce output here, and regular C++ comments which are ignored by doxygen, and look at the source code to see how line breaks in this paragraph are produced.

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Chapter 4

Class Documentation

4.1 message_cls Class Reference

`#include <message_cls.h>`

Public Member Functions

- void pt (const char *text) const
- void pi (const int num) const
- void pti (const char *text, const int num) const
- void ptf (const char *text, const float num) const
- void chk (const int) const

Private Attributes

- char * text

4.1.1 Member Function Documentation

4.1.1.1 chk()

```
void message_cls::chk (
            const int num ) const
```

prints the checkpoint location: checkpoint —.

Returns

void

Parameters

num	checkpoint.

Precondition

1. int checkpoint be printed.

Postcondition

1. no output.

4.1.1.2 pi()

```
void message_cls::pi (
            const int num ) const
```

prints an integer.

Returns

void

Parameters

num	an int variable.

Precondition

1. int number to be printed.

Postcondition

1. no output.

4.1.1.3 pt()

```
void message_cls::pt (
            const char * text ) const
```

prints a text.

Returns

void

Parameters

text	a string to be printed.

Precondition

1. text contains a string to be printed.

Postcondition

1. no output.

4.1.1.4 ptf()

```
void message_cls::ptf (
            const char * text,
            const float num ) const
```

prints a string followed by a float.

Returns

void

Parameters

text	a char variable. num a float variable.

Precondition

1. float number to be printed.

Postcondition

1. no output.

4.1.1.5 pti()

```
void message_cls::pti (
            const char * text,
            const int num ) const
```

prints a string followed by an integer .

Returns

void

Parameters

text	a char variable. num an int variable.

Precondition

1. int number to be printed.

Postcondition

1. no output.

4.1.2 Member Data Documentation

4.1.2.1 text

```
char* message_cls::text  [private]
```

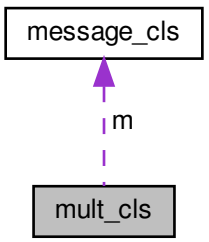
The documentation for this class was generated from the following files:

- include/module_1/message_cls.h
- src/module_1/message_cls.cpp

4.2 mult_cls Class Reference

```
#include <mult_cls.h>
```

Collaboration diagram for mult_cls:



Public Member Functions

- mult_cls (const int)
- int doit (const int, const int)
- float doit (const float, const float)
- int vecXvec (std::vector< int >, std::vector< int >)

Public Attributes

- int irestults
- float frestults

Private Member Functions

- mult_cls ()

Private Attributes

- message_cls m
- int input
- int ivar1
- int ivar2
- int ires = 0
- float fvar1
- float fvar2
- float fres = 0
- std::vector< int > vec1
- std::vector< int > vec2

4.2.1 Constructor & Destructor Documentation

4.2.1.1 mult_cls() [1/2]

```
mult_cls::mult_cls ( )  [private]
```

Here is the call graph for this function:

4.2.1.2 mult_cls() [2/2]

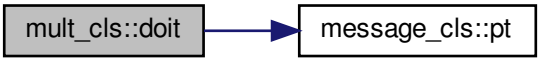
```
mult_cls::mult_cls (
            const int input )
```

4.2.2 Member Function Documentation

4.2.2.1 doit() [1/2]

```
float mult_cls::doit (
            const float fvar1,
            const float fvar2 )
```

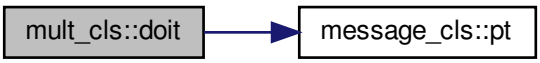
Here is the call graph for this function:



4.2.2.2 doit() [2/2]

```
int mult_cls::doit (
            const int ivar1,
            const int ivar2 )
```

Here is the call graph for this function:



4.2.2.3 vecXvec()

```
int mult_cls::vecXvec (
            std::vector< int > vec1,
            std::vector< int > vec2 )
```

Here is the call graph for this function:

4.2.3 Member Data Documentation

4.2.3.1 fres

```
float mult_cls::fres = 0  [private]
```

4.2.3.2 frestults

```
float mult_cls::frestults
```

4.2.3.3 fvar1

```
float mult_cls::fvar1  [private]
```

4.2.3.4 fvar2

```
float mult_cls::fvar2  [private]
```

4.2.3.5 input

```
int mult_cls::input  [private]
```

4.2.3.6 ires

```
int mult_cls::ires = 0  [private]
```

4.2.3.7 irestults

```
int mult_cls::irestults
```

4.2.3.8 ivar1

```
int mult_cls::ivar1  [private]
```

4.2.3.9 ivar2

```
int mult_cls::ivar2  [private]
```

4.2.3.10 m

```
message_cls mult_cls::m  [private]
```

4.2.3.11 vec1

```
std::vector<int> mult_cls::vec1  [private]
```

4.2.3.12 vec2

```
std::vector<int> mult_cls::vec2  [private]
```

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

- include/module_1/mult_cls.h
- src/module_1/mult_cls.cpp

Chapter 5

File Documentation

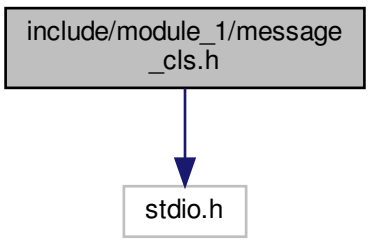
5.1 include/module_1/division_cls.h File Reference

5.2 include/module_1/message_cls.h File Reference

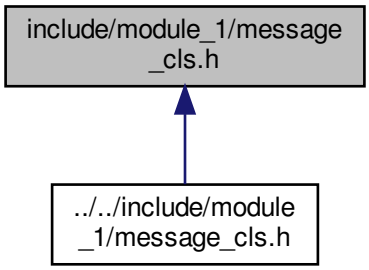
A class for all print functions.

```
#include <stdio.h>
```

Include dependency graph for message_cls.h:



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



Classes

- class message_cls

5.2.1 Detailed Description

A class for all print functions.

Author

Poursartip:Babak:PhD:Algo

Version

Revision 1.1

pt: prints a text message pi: prints an int number pi: prints a text and an int number chk: checkpoint printer

Date

Monday, July 20, 2020

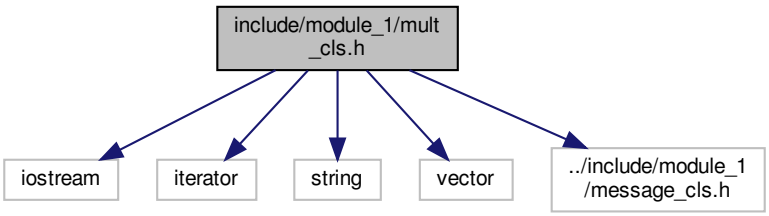
5.3 include/module_1/mult_cls.h File Reference

it does number multiplication

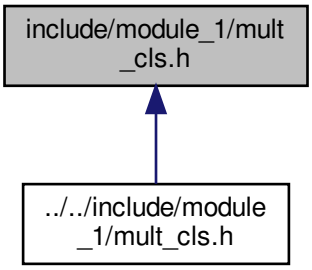
```
#include <iostream>
#include <iterator>
#include <string>
#include <vector>
```

```
#include "../include/module_1/message_cls.h"
```

Include dependency graph for mult_cls.h:



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



Classes

- class mult_cls

5.3.1 Detailed Description

it does number multiplication

Author

Poursartip:Babak:PhD:mult

Version

Revision 1.20

mult_int_int: multiplies int by int mult_float_float: multiplies float by float mult_vector_vector: multiplies float by float

Date

Monday, July 20, 2020

5.4 input/clustering1.txt File Reference

5.5 input/clustering_test.txt File Reference

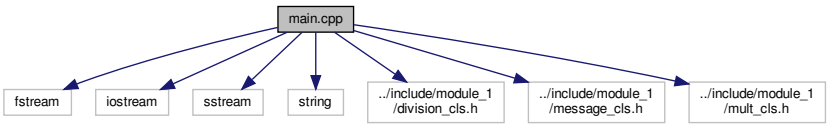
5.6 input/clustering_test2.txt File Reference

5.7 main.cpp File Reference

Illustrates doxygen-style comments for documenting a C++ program file and the functions in that file.

```
#include <fstream>
#include <iostream>
#include <sstream>
#include <string>
#include "../include/module_1/division_cls.h"
#include "../include/module_1/message_cls.h"
#include "../include/module_1/mult_cls.h"
```

Include dependency graph for main.cpp:



Functions

- int main ()

5.7.1 Detailed Description

Illustrates doxygen-style comments for documenting a C++ program file and the functions in that file.

Author

Poursartip:Babak:PhD:Algo

Version

Revision 1.1

If you want to add any further detailed description of what is in the file, then place it here (after the first statement) and it will appear in the detailed description section of the HTML output description for the file.

Date

Monday, July 20, 2020

5.7.2 Function Documentation

5.7.2.1 main()

```
int main ( )
```

< Compute and return the number of digits in a positive integer. The style used for the pre/post coditions of this function is purposely different from that for the void function given above, just so you can see the differnce in the HTML output.

Returns

The number of digits in n.

Parameters

n	An integer, the number of whose digits is desired.

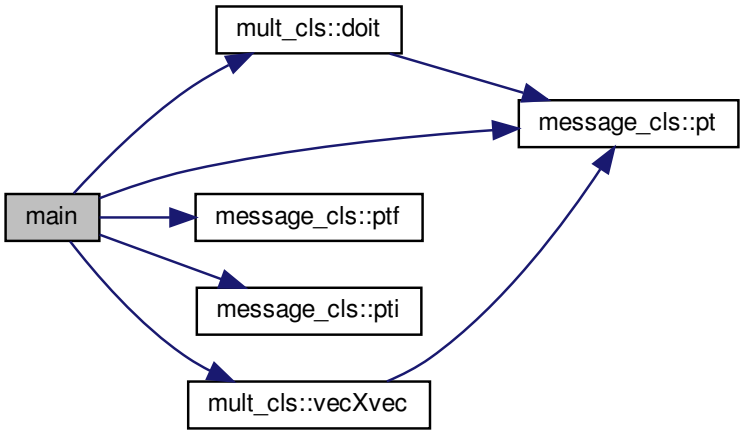
Precondition

n contains a positive integer. This is some other precondition, and note that it does not start on a new line.

Postcondition

The number of digits in n has been returned.
This is some other post condition, and note that it does start on a new line.

Here is the call graph for this function:



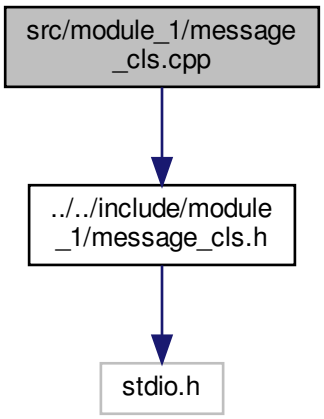
5.8 src/module_1/division_cls.cpp File Reference

5.9 src/module_1/message_cls.cpp File Reference

A class for all print functions.

```
#include "../../include/module_1/message_cls.h"
```

Include dependency graph for message_cls.cpp:



5.9.1 Detailed Description

A class for all print functions.

Author

Poursartip:Babak:PhD:Algo

Version

Revision 1.1

pt: prints a text message pi: prints an int number pi: prints a text and an int number chk: checkpoint printer

Date

Monday, July 20, 2020

5.10 src/module_1/mult_cls.cpp File Reference

```
#include "../../include/module_1/mult_cls.h"
```

Include dependency graph for mult_cls.cpp: