Documenting the code using doxygen

In this exercise we will use doxygen to create a minimal documentation for the Jacobi code.

C

Let's first generate the configuration file

```
doxygen -g
```

Edit the file and modify

```
PROJECT_NAME           = "Jacobi"
PROJECT_BRIEF          = "Documentation for Jacobi Solver"
OUTPUT_DIRECTORY       = docs
```

Then, modify all the beginnings of each source file using doxygen markup, in prarticular

```
/brief
/details
/author
/copyright
```

Other tags that can be used are \version, \date, \pre, \bug, \warning Merge the licence information into the copyright field. Remember that the doxygen comment block has to start with /**

Also, add the field \file at the very beginning, using the file name

```
/** \file jacobi_function.c
```

to have the file listed and browsable in the doc.

Running

```
doxygen
```

should generate the documentation in the docs folder. Open with a browser the file

```
docs/html/index.html
```

Next, document the methods in jacobi_functions.h. You have to put the description BEFORE the function declaration. For example

```
/** Prints the output in text format
  * \param[in] grid double precision array containing the first grid buffer
  * \param[in] dimension the size of the system. The buffers should be of size (dimension + 2)^2
  */
void printOutput(double* grid, int dimension);
```

Do the same for update and setBoundaryConditions. Then generate the documentation with

```
doxygen
```

and browse it.

Finally, let's add a starting page. To this end, add a \mainpage section to one of the file (jacobi_main.c is a good choice)

```
/** \mainpage Jacobi solver documentation
 *
 *  \section intro_sec Introduction
 *           Welcome to the documentation for the Laplace solver. This program is written in C and it ROCKS!  BIG TIME!
 *  \authors Giuseppe Piero Brandino - eXact-lab s.r.l.
 *  \copyright 2019 eXact-lab s.r.l. GPLv2
 */
```

Generate the final version of your docs!

```
doxygen
```

Alternatively, you can feed a markdown file as a main page

```
USE_MDFILE_AS_MAINPAGE = "main.md"
```

Python

For python, we will be using a combination of doxygen and docstrings. Docstrings is the built-in documentation module for the creation of help functions to be use in the python console.

Let's first generate the configuration file

```
doxygen -g
```

Edit the file and modify

```
PROJECT_NAME           = "Jacobi"
PROJECT_BRIEF          = "Documentation for Jacobi Solver"
OUTPUT_DIRECTORY       = docs
```

Then, modify all the beginnings of each source file using doxygen markup, in prarticular

```
/brief
/details
/author
/copyright
```

Other tags that can be used are \version, \date, \pre, \bug, \warning Merge the licence information into the copyright field. Remember that the doxygen comment block has to start with ##

Also, add the field \file at the very beginning, using the file name

```
## \file jacobi_function.c
```

to have the file listed and browsable in the doc.

Running

```
doxygen
```

should generate the documentation in the docs folder. Open with a browser the file

```
docs/html/index.html
```

Next, document the methods in jacobi_functions.py. For python we will be using a combination of doxygen and docstrings. Namely, we will be putting docstrings-style description after the function declaration For example

```
def printOutput(grid, dimension):
    """ Prints the output in text format
    Args:
        grid double precision array containing the first grid buffer
        dimension the size of the system. The buffers should be of size (dimension + 2)^2
    """
```

Do the same for update and setBoundaryConditions. Then generate the documentation with

```
doxygen
```

and browse it. The documentation for the function is less fancy now. However, the advantage is that if we use the python console and load the module

```
python
import jacobi_functions as jacobi
```

We can get the function documentation in the console as well

```
help(jacobi.printOutput)
```

Finally, let's add a starting page. To this end, add a \mainpage section to one of the file (jacobi_main.py is a good choice)

```
## \mainpage Jacobi solver documentation
#
#  \section intro_sec Introduction
#           Welcome to the documentation for the Laplace solver. This program is written in C and it ROCKS!  BIG TIME!
#  \authors Giuseppe Piero Brandino - eXact-lab s.r.l.
#  \copyright 2019 eXact-lab s.r.l. GPLv2
#
```

Generate the final version of your docs!

```
doxygen
```

Alternatively, you can feed a markdown file as a main page

```
USE_MDFILE_AS_MAINPAGE = "main.md"
```