

# Documentation (PHY1610 Lecture 4)

Ramses van Zon   Marcelo Ponce

Winter 2021

## Section 1

# Documentation

# Document your modules

The most unlikely pieces of code can end up being reused, so try and add at least a bit of documentation.

There are many documentation styles and philosophies:

- **No documentation**

This style also often advocates **no comments**. The pretense is that clear code is 'self-documenting'.

Sure, but . . . yeah, sorry, no.

- **Auto-generated documentation**

Adding specially formatted comments that a tool like **doxygen** uses to generate documentation.

This is pretty decent, and a good way to keep documentation up-to-date when the code changes.

- **New-user oriented**

Your code will get read by someone with (much) less understanding of what it's supposed to do than you. If you were in this situation, what documentation would you need to be able to use the module?

If you do nothing else, at least add a README.md file.

# Doxygen (an example)

```
/// @file  outputarray.h
/// @author Ramses van Zon
/// @date   January 14, 2019
/// @brief  Module for writing a 1d array of doubles to text and binary files.
#ifndef OUTPUTARRAYH
#define OUTPUTARRAYH
#include <string>

/// @brief Function to write an array of doubles to a binary file.
/// This function does a raw dump of the array file to file.
/// @param s  the filename
/// @param n  number of elements of the array to write to file
/// @param x  pointer to the first element of the array of doubles
void writeBinary(const std::string& s, int n, const double x[]);

/// @brief Function to write an array of doubles to a text file.
/// The file will contain each element of the array on a separate line.
/// @param s  the filename
/// @param n  number of elements of the array to write to file
/// @param x  pointer to the first element of the array of doubles
void writeText(const std::string& s, int n, const double x[]);
#endif
```

# Doxygen (an example, continued)

- 1 Generate a configuration file for doxygen called Doxygen (then edit it):

```
doxygen -g
sed -i 's/PROJECT_NAME[ ]*=.*\/PROJECT_NAME=Outputarray/' Doxyfile
```

- 2 Create a README.md

```
[//]: # \mainpage
Outputarray is a module for writing a 1d array of doubles to text and binary files.
Compile with: "g++ -c -std=c++11 outputarray.cc -o outputarray.o"
Generate documentation with doxygen as follows
```

```
doxygen -g
sed -i 's/PROJECT_NAME[ ]*=.*\/PROJECT_NAME=Outputarray/' Doxyfile
doxygen
make -C latex
```

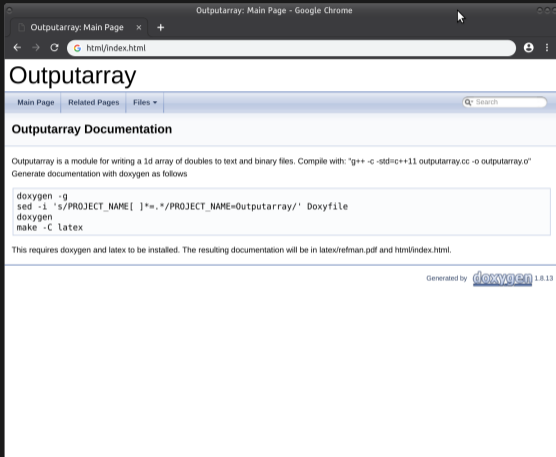
This requires doxygen and latex to be installed.

The resulting documentation will be in latex/refman.pdf and html/index.html.

- 3 Generate the documentation in html and latex form:

```
$ doxygen
```

# Doxygen (an example, html result)



Outputarray: Main Page - Google Chrome

Outputarray: Main Page x +

html/index.html

## Outputarray

Main Page Related Pages Files ▾

Search

### Outputarray Documentation

Outputarray is a module for writing a 1d array of doubles to text and binary files. Compile with: "g++ -c -std=c++11 outputarray.cc -o outputarray.o"  
Generate documentation with doxygen as follows

```
doxygen -g
sed -i 's/PROJECT_NAME[ ]*=./PROJECT_NAME=Outputarray/' Doxyfile
doxygen
make -C latex
```

This requires doxygen and latex to be installed. The resulting documentation will be in latex/refman.pdf and html/index.html.

Generated by [doxygen](#) 1.8.13



Outputarray: outputarray.h File Reference - Google Chrome

Outputarray: outputarray x +

outputarray\_8h.html#ac41a5e009bbd1f3f911dea658a504ae9

### Function Documentation

#### ◆ writeBinary()

```
void writeBinary ( const char * s,
                  double * x,
                  int n
                  )
```

Function to write an array of doubles to a binary file.

This function does a raw dump of the array file to file.

**Parameters**

- s the filename
- x pointer to the first element of the array of doubles
- n number of elements of the array to write to file

#### ◆ writeText()

```
void writeText ( const char * s,
                double * x,
                int n
                )
```

Function to write an array of doubles to a text file.

The file will contain each element of the array on a separate line.