Doxygen

Kameswari Chebrolu

https://images-cdn.9gag.com/photo/aGg2wQX_460s.jpg

Why Document?

- Better understanding of code, especially in big teams
  - By self after some time or by team-mates
- Required by company / administrators who examine code
  - For quality control, getting certification
- Why automation via Doxygen?
  - Better Structure, easy to read
  - Different formats (html, rtf, latex, pdf etc)

Doxygen

- Doxygen is a documentation generator for C++, C, Java, Python, PHP etc
  - Most popular for C++
  - Javadoc is more popular for Java and Sphinx/Pydoc for python
- Supports automated visualization also
  - Relation between various elements by means of include dependency graphs, inheritance diagrams, and collaboration diagrams
- Highly portable
  - Supported in Windows/unix/mac

Steps

- Create a config file
  - doxygen –g <file-name>
- Open file and fill in the blanks
  - PROJECT_NAME = MyProject
  - OUTPUT_DIRECTORY = ./doc
  - INPUT = ./src ./include
  - FILE_PATTERNS = *.cpp *.hpp
  - GENERATE_HTML = YES
  - EXTRACT_ALL = YES
  - Etc
- Comment your code
- "doxygen [configfile]" will create the documentation as per specification

Commenting

- Two types of comments:
  - Comments at the beginning of each file
    - Describes the file, author, known bugs etc
  - Comments that describe functions and data structures etc

Do’s and Don’ts

- Each file/function should have a header block
- Use descriptive and meaningful names for variables, constants, and functions
- Don't just re-express the algorithm in English
  - Right: For each name in the array, extract the last name.
  - Wrong: Set i to 0. Loop from 0 to 10. Call strchr() on a[i] etc

Important Tags

- @file: Each file needs to begin with this, followed by the name of the file
- @brief: brief description of the file
  - Can be followed by a more detailed description
- @author: Used to specify names of the authors
- @bug: List of known bugs
  - Even if there are no bugs, state it explicitly with @bug tag

Important Tags

- Can comment functions in either
  - header files where they are declared or
  - source files where they are implemented
- @brief: comment block to give a description of the function
- @param: use to describe all of the parameters of the function
- @return: use to describe the return value of the function

References

https://doxygen.nl/manual/commands.html (list of all commands)

https://www.mitk.org/images/1/1c/BugSquashingSeminars%242013-07-17-DoxyReference.pdf (quick reference)