CS100
Introduction to Programming

Lecture 28 Doxygen

Introduction

- Doxygen is a documentation system for C++, C, Java, etc.
- It helps in
  - Generating on-line documentation (web-page) or offline reference manual (PDF) from documented source files
  - Extracting the code structure and visualising the relations between various elements by means of include dependency graphs, inheritance diagrams, and collaboration diagrams, which are all generated automatically

Steps

1. Create a Configuration File
2. Document the Code
3. Run the Doxygen

Configuration file

- Doxygen determines settings from configuration file
- Example name: Doxyfile
- Doxywizard is a GUI front-end for configuring and running doxygen

Example configuration file

- See:
  - https://github.com/laurentkneip/opengv/blob/master/Doxyfile
- Important setting:
  - PROJECT_NAME = “…”
  - PROJECT_BRIEF = “…”
  - OUTPUT_DIRECTORY = “doc”
  - INPUT = “include”
    INPUT += “doc/addons”
  - EXCLUDE_PATTERNS = “*/implementation/*”
    EXCLUDE_PATTERNS += “*/modules/*”
  - IMAGE_PATH = “doc/addons/images”
  - GENERATE_HTML = YES

Documenting Code

- Document Blocks or Lines
- Document members
- Structural Commands
- Create Lists

Document Block

- Specify Comment Blocks
  - Special documentation blocks , Comment Blocks with additional markings
  - Brief and Detailed Description

Detailed Description

- Extra * or ! For Comment Blocks

```
/**
 * ... text ...
 */
```

- or

```
/*!
 * ... text ...
 */
```

Detailed Description

- For Comment Lines-additional slash or an exclamation mark e.g.

```
///
/// ... text ...
///
```

- or

```
//!
//!... text ...
//!
```

Brief Description

- Command: `\brief`

- Syntax:

```
/*! \brief Brief description.
 *         Brief description continued.
 *
 *   Detailed description starts here.
 */
```

Document Members

- Documentation after member specification in source file.
- Use ‘<‘ , that indicates that member is located in front of block
- Syntax:

```
int var ; /*!< Description   */
```

- or brief (one-line) Description

```
int var ; //!< Brief Description
```

Structural Commands

- Starts with ‘\’ or ‘@’
- Main commands

```
\author {list of authors}
\brief {brief description}
\date {date description}
\file [<name>]
\fn (function name)- used if comment block is not placed
\param <parameter-name> {parameter description}
\return {return value description}
```

Structural Commands

- Visual Enhancement Commands

```
\a - Special font
\b - bold
\c - Typewriter Font
\arg - Simple bulleted list, not nested
\e  - Italics
\em - Emphasize word and in Italics
```

Creating Lists

- Bulleted Lists
  - Column aligned, using Minus Sign ‘-’
- Numbered Lists
  - Column aligned, using Minus Sign followed by Hash ‘-#’
- Nesting Level is maintained according to column alignment
- HTML Commands
  - Can be used inside comment blocks

Creating Lists

- Syntax:

```
/*!
 * List of events
 *   -Event1
 *      -#Subevent1
 *      -#Subevent2\n
 *        subevent2 cont.
 *      -#Subevent3
 *   -Event2
 *   -Event3
 */
```

Example

- OpenGV:

  https://github.com/laurentkneip/opengv/blob/master/include/opengv/absolute_pose/methods.hpp