

Генераторы документации по исходным текстам

Пример использования Doxygen
см. в демонстрационном проекте StringSearch к ЛР № 2

Кому, когда и зачем нужна документация?

- Разработчикам
 - Анализ программного интерфейса
 - Какие сущности (функции, типы данных) входят в код?
 - Каким *в точности* заявлено поведение функций?
 - Какова структура кода?
- Руководителям в ИТ
 - Как часть отчетности по проекту
 - Принятие решения об использовании библиотеки
 - Определение круга задач, решаемых библиотекой
 - Оценка стоимости внедрения
 - Объем и сложность структуры библиотеки
 - Наличие и качество документации (*замкнутый круг*)

Способы получения сведений о программе

Написание документации вручную

- Логичная структура
- Перекрестные ссылки, оглавление, поиск, иллюстрации
- Эргономичное оформление
- Высокая трудоемкость
 - Нужен технический писатель
- Необходимость актуализации при изменениях кода

Чтение исходных текстов

- Самая актуальная информация
- Комментарии уже имеются
- Нужна квалификация
 - Сложно для руководителя
 - Надолго отвлекает разработчика
- Структура диктуется техническими соображениями, часто неудобна
 - Трудно получить общее представление
- Выразительного оформления нет

Решение — генераторы документации

- По исходному коду можно сформировать документацию
 - Оглавление, алфавитный указатель, поиск
 - Диаграммы (структура библиотеки)
- В коде есть комментарии
 - Их можно включить в документацию
 - В них можно указывать команды генератору документации
 - Управление форматированием
 - Перекрестные ссылки
 - Частично генерируются автоматически
 - Разнесение сущностей по логическим разделам

Свойства генерируемой документации

- По качеству структуры и оформления — как рукописная
 - Несколько меньшая гибкость
 - Приходится вносить много информации в комментарии
 - Код становится труднее читать
- Актуализируется автоматически
- Малая трудоемкость
 - Достаточно поддерживать комментарии в коде
 - Желательно соблюдать формат комментариев
 - Можно формировать документы, файлы справки, версии для web
- Привычный формат

Популярные генераторы документации

- Doxygen (C-подобные языки)
 - DoxyBlocks — расширение Code::Blocks

Индустриальный стандарт для C++
- JavaDoc (Java)

Индустриальный стандарт для Java

 - PhpDocumentor (PHP), JSDoc (Java Script)
- Sphinx (Python и другие)
 - Python имеет встроенные в язык `"""docstrings"""`
- Doc-O-Matic

Использование Doxygen в Code::Blocks

- Установить Doxygen
- Установить расширение DoxyBlocks
 - Устанавливается по умолчанию

- Настроить DoxyBlocks:

DoxyBlocks →  *Open preferences...* →

- Для генерации документации:

DoxyBlocks →  *Extract documentation*

- Для просмотра:

DoxyBlocks →  *Run HTML*

