

MakeHarrixMathLibrary v.2.1

А. Б. Сергиенко

5 июня 2014 г.

Аннотация

MakeHarrixMathLibrary — это программа собирающая библиотеку HarrixMathLibrary и справку к ней из исходных материалов.

Содержание

1	Внешний вид программы	2
2	Результат работы программы	2
3	Как собирается библиотека	3
4	Как собирается справка	4
5	Исходники MakeHarrixMathLibrary.exe и справки по ним	6

1 Внешний вид программы

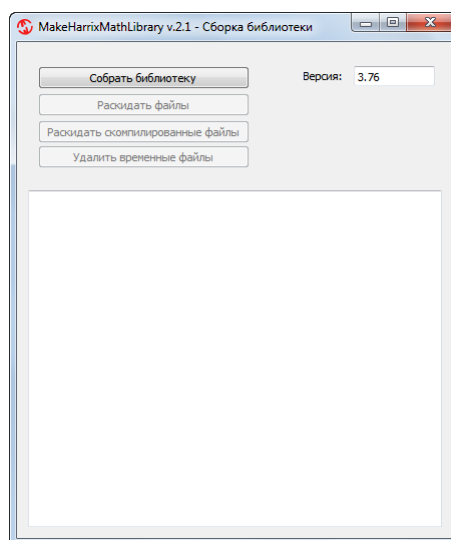


Рисунок 1. Внешний вид программы

При нажатии на кнопку **«Собрать библиотеку»** будет производиться сборка библиотеки вместе с файлами справки. После чего будет открыта папка с сформированными файлами.

В поле для ввода вы можете ввести версию библиотеки, которую вы собираете. В собираемых файлах данный номер проставится автоматически. Исключение составляют файлы README.md и CHANGELOG.md. Но это и логично.

В текстовом поле под кнопкой будет отображаться ход работы программы.

2 Результат работы программы

В папке **source_library** находится исходный материал, который обрабатывается программой MakeHarrixMathLibrary.exe, в результате чего образуются следующие элементы:

- **HarrixMathLibrary.cpp** — главный файл библиотеки;
- **HarrixMathLibrary.h** — заголовочный файл;
- **HarrixMathLibrary_Help.tex** — файл справки в формате \LaTeX .
- **FUNCTIONS.md** — файл со списком функций.
- **mtrand.cpp** — исходный код для генератора случайных чисел Mersenne Twister;
- **mtrand.h** — заголовочный файл для генератора случайных чисел Mersenne Twister;
- **names.tex** — дополнительный файл для HarrixMathLibrary_Help.tex;
- **packages.tex** — дополнительный файл для HarrixMathLibrary_Help.tex;
- **styles.tex** — дополнительный файл для HarrixMathLibrary_Help.tex;

- **biblio.bib** — файл о библиографических источниках, которые могут быть использованы в справке;
- **utf8gost705u.bst** — файл оформления справки;
- **\images** — папка с рисунками форматов *.png или *.pdf для HarrixMathLibrary_Help.tex.

Все данные файлы собираются в папке **temp_library**.

Описание того, что делать с полученными файлами описано в разделе «**Как добавлять свои функции**» в файле HarrixMathLibrary_Help.pdf в папке **_library**.

3 Как собирается библиотека

Исходники библиотеки находятся в папке **source_library**.

Файлы **HarrixMathLibrary.cpp** и **HarrixMathLibrary.h** собираются следующим образом.

Итак, вначале добавляются к файлам некоторые основные файлы:

- **Header.cpp** — основная информация, подключение заголовочных файлов (в нем текст **[VERSION]** заменится на номер версии, который вы указываете при сборке);
- **AdditionalVariables.cpp** — содержит список дополнительных переменных, которые используются внутри функций. Самим использовать их не нужно — это только внутренние переменные;
- **Random.cpp** — две основные функции для работы со случайными числами: **MHL_SeedRandom()** и **MHL_RandomNumber()**;
- **Random.h** — объявления этих самых функций: **MHL_SeedRandom()** и **MHL_RandomNumber()**;
- **Const.h** — содержит список констант для работы с библиотеки;
- **Enum.h** — переменные перечисляемого типа.

Также **HarrixMathLibrary.h** обрамляется кодом:

Код 1. Обрамление HarrixMathLibrary.h файла

```
#ifndef HARRIXMATHLIBRARY_H
#define HARRIXMATHLIBRARY_H

#endif // HARRIXMATHLIBRARY_H
```

Также в папке **source_library** есть директории. Каждая директория — это множество функций какого-то раздела. Перед рассмотрением файлов папки программа добавляет в файл **HarrixMathLibrary.cpp** следующий код:

Код 2. Название раздела

```
//*****
//[Название папки]
//*****
```

А в файл `HarrixMathLibrary.h` добавляется код:

```
Код 3. Название раздела
```

```
//[Название папки]
```

После каждой функции в `HarrixMathLibrary.cpp` вставляется код:

```
Код 4. Название раздела
```

```
//-----
```

Далее программа пробегает по каждой папке, которая представляет собой раздел функций в библиотеке. Каждая функция в разделе предоставляется следующими файлами:

- **<File>.cpp** или **<File>.tpp** — код функции;
- **<File>.h** — заголовочный файл функции;
- **<File>.tex** — справка по функции;
- **<File>.desc** — описание функции;
- **<File>.use** — пример использования функции (из него удаляются пробелы в начале строк, равным числу пробелов в начале первой строки);
- **<File>_<name>.pdf** — множество рисунков, необходимых для справки по функции (необязательные файлы);
- **<File>_<name>.png** — множество рисунков, необходимых для справки по функции (необязательные файлы);

Важно помнить, что каждый *.cpp, *.h, *.tex файл в папках папки **source_library** не является полноценным файлом соответствующего расширения и без сборки в единые файлы библиотеки не может использоваться.

Разница файлов *.cpp и *.tpp в том, что в *.tpp пишется код шаблонов функций, а в *.cpp пишутся обычные функции, и их реализация располагается в `HarrixMathLibrary.cpp` файле, тогда как шаблоны располагаются в `HarrixMathLibrary.h` файле.

Ниже показан алгоритм (Алгоритм .1) формирования файлов библиотеки.

Итоговое количество функций определяется как количество знаков «;» в h файлах функций, которые располагаются в папках.

Стоит отметить, что все разделы функций и сами функции сортируются в алфавитном порядке.

Файл **FUNCTIONS.md** собирается аналогичным образом.

4 Как собирается справка

Исходники файлов справки библиотеки находятся с папке **source_library**.

Файлы **HarrixMathLibrary.tex** собирается следующим образом.

Итак, вначале добавляются к файлу некоторые основные файлы:

Алгоритм .1. Алгоритм собирания файлов библиотеки

Начало алгоритма

HarrixMathLibrary.cpp+ = Header.cpp;
HarrixMathLibrary.cpp+ = AdditionalVariables.cpp;
HarrixMathLibrary.cpp+ = Random.cpp;
HarrixMathLibrary.h+ = Начало обрамления;
HarrixMathLibrary.h+ = Const.h;
HarrixMathLibrary.h+ = Random.h;
HarrixMathLibrary.h+ = Enum.h;

Выполнить для всех папок выполнять

HarrixMathLibrary.cpp+ = Код 1. Название раздела;

HarrixMathLibrary.h+ = Код 2. Название раздела;

Выполнить для всех файлов папки расширения *.cpp, *.tpp и *.h выполнять**Если есть файл *.cpp тогда**

HarrixMathLibrary.cpp+ =< File > .cpp;

иначе

ResultTpp+ =< File > .tpp;

Конец условия

HarrixMathLibrary.h+ =< File > .h;

Конец цикла**Конец цикла**

HarrixMathLibrary.h+ = ResultTpp;

HarrixMathLibrary.h+ = Конец обрамления;

Сохранить *HarrixMathLibrary.cpp* в папке temp_library;

Сохранить *HarrixMathLibrary.h* в папке temp_library;

Конец алгоритма

- **Title.tex** — титульная информация и содержание справки(в нем текст **[VERSION]** заменится на номер версии, который вы указываете при сборке);
- **Description_part1.tex** и **Description_part2.tex** — описание библиотеки (разделено на два файла, чтобы между ними вставить число функций);
- **Install.tex** — содержит информацию об установке и использованию библиотеки;
- **Random.tex** — информация о случайных числах в библиотеке;
- **Addnew.tex** — информация о том, как добавлять новые функции в библиотеку.
- **Thirdparty.tex** — информация о сторонних библиотеках, используемых в HarrixMathLibrary.

Ниже показан алгоритм (Алгоритм .2) формирования файлов справки библиотеки.

Некоторые моменты по преобразованию некоторых данных (например, преобразование `<File>.desc`) не рассматривается в алгоритме, но вы можете все посмотреть в исходном коде программы, которая поставляется с данной библиотекой в папке **source_make**

5 Исходники MakeHarrixMathLibrary.exe и справки по ним

MakeHarrixMathLibrary написан на Qt. Не требует каких-то дополнительных файлов. Исходники программы располагаются в папке **source_make**.

Исходники справки MakeHarrixMathLibrary (данного файла, который вы читаете) по располагаются в папке **source_make\help**. Главный файл источника справки — это файл **MakeHarrixMathLibrary_Help.tex**.

Алгоритм .2. Алгоритм собирания файлов справки библиотеки

Начало алгоритма

HarrixMathLibrary_Help.tex + = *Install.tex*;
HarrixMathLibrary_Help.tex + = *Random.tex*;
HarrixMathLibrary_Help.tex + = *Addnew.tex*;
HarrixMathLibrary_Help.tex + = *Thirdparty.tex*;
ResultTexList + = Заголовок для списка функций;
ResultTexFunctions + = Заголовок для функций;

Выполнить для всех папок выполнять

ResultTexList + = Заголовок раздела;
ResultTexFunctions + = Заголовок раздела;
n = 0;

Выполнить для всех файлов папки расширения *.desc, *.tex, *.h, *.use выполнять

ResultTexList + = < *File* > .desc в обработке;
ResultTexFunctions + = < *File* > .desc в обработке;
ResultTexFunctions + = < *File* > .h в обработке;
ResultTexFunctions + = < *File* > .tex;
ResultTexFunctions + = < *File* > .use в обработке;
n + +;

Конец цикла**Выполнить для всех файлов папки расширения *.pdf и *.png выполнять**

Скопировать файл <File>.<png|pdf> в папку \images\;

Конец цикла**Конец цикла**

HarrixMathLibrary_Help.tex + = *ResultTexList*;
HarrixMathLibrary_Help.tex + = *ResultTexFunctions*;
HarrixMathLibrary_Help.tex + = Концовка документа;
HarrixMathLibrary_Help.tex = *Description_part2.tex* + *HarrixMathLibrary_Help.tex*;
HarrixMathLibrary_Help.tex = *n* + *HarrixMathLibrary_Help.tex*;
HarrixMathLibrary_Help.tex = *Description_part1.tex* + *HarrixMathLibrary_Help.tex*;
HarrixMathLibrary_Help.tex = *Title.tex* + *HarrixMathLibrary_Help.tex*;
HarrixMathLibrary_Help.tex = Список литературы;
Сохранить *HarrixMathLibrary_Help.tex* в папке temp_library;

Конец алгоритма
