

Thread Pool+Archiver

0.3

Создано системой Doxygen 1.8.15

1 Иерархический список классов	1
1.1 Иерархия классов	1
2 Алфавитный указатель классов	2
2.1 Классы	2
3 Классы	2
3.1 Структура bi_file	2
3.2 Структура dictionary	3
3.3 Класс lib_archiver	3
3.4 Класс LZW_archiver	3
3.5 Класс own_archiver	4
3.6 Структура qlz_hash_compress	4
3.7 Структура qlz_hash_decompress	4
3.8 Структура qlz_state_compress	5
3.9 Структура qlz_state_decompress	5
3.10 Класс Tar	5
3.11 Структура tarheader	6
3.12 Структура task	6
3.13 Класс TaskQueue	7
3.14 Класс Testing_class	7
3.15 Класс Thread_pool	8
Предметный указатель	9

1 Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

bi_file	2
dictionary	3
LZW_archiver	3
qlz_hash_compress	4
qlz_hash_decompress	4
qlz_state_compress	5
qlz_state_decompress	5
Tar	5
lib_archiver	3
own_archiver	4

tarheader	6
task	6
TaskQueue	7
Testing_class	7
Thread_pool	8

2 Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

bi_file	2
dictionary	3
lib_archiver	3
LZW_archiver	3
own_archiver	4
qlz_hash_compress	4
qlz_hash_decompress	4
qlz_state_compress	5
qlz_state_decompress	5
Tar	5
tarheader	6
task	6
TaskQueue	7
Testing_class	7
Thread_pool	8

3 Классы

3.1 Структура bi_file

Открытые атрибуты

- FILE * file
- uchar mask
- int rack
- int pacifier_counter

Объявления и описания членов структуры находятся в файле:

- Thread_Pool+Archiver/own_archiver.h

3.2 Структура dictionary

Открытые атрибуты

- int code_value
- int prefix_code
- char character

Объявления и описания членов структуры находятся в файле:

- Thread_Pool+Archiver/own_archiver.cpp

3.3 Класс lib_archiver

Базовые классы: [Tar](#).

Открытые члены

- int compress (const char *in, const char *out) override
- int decompress (const char *in, const char *out) override

Объявления и описания членов классов находятся в файлах:

- Thread_Pool+Archiver/libarch.h
- Thread_Pool+Archiver/libarch.cpp

3.4 Класс LZW_archiver

Открытые члены

- LZW_archiver (const char *filename, int mode)
- void close ()
- void add_to_archive (const char *filename)
- void untar (FILE *a, const char *name)
- int extract (const char *tarFileName)
- int verify_checksum (const char *p)
- FILE * create_file (char *name)
- int is_end_of_archive (const char *p)
- int parseoct (const char *p, size_t n)

Объявления и описания членов классов находятся в файлах:

- Thread_Pool+Archiver/archive.h
- Thread_Pool+Archiver/archiver.cpp

3.5 Класс `own_archiver`

Базовые классы: [Tar](#).

Открытые члены

- `int compress (const char *in, const char *out) override`
- `int decompress (const char *in, const char *out) override`
- `std::shared_ptr< bi_file > Open_File (const char *name, const char *mode)`
- `void Close_File (std::shared_ptr< bi_file > b, int mode)`
- `void WriteBits (std::shared_ptr< bi_file > bfile, ulong code, int count)`
- `ulong ReadBits (std::shared_ptr< bi_file > bfile, int bit_count)`
- `uint find_dictionary_match (int prefix_code, int character)`
- `uint decode_string (uint count, uint code)`

Открытые атрибуты

- `char decode_stack [TABLE_SIZE]`

Объявления и описания членов классов находятся в файлах:

- `Thread_Pool+Archiver/own_archiver.h`
- `Thread_Pool+Archiver/own_archiver.cpp`

3.6 Структура `qlz_hash_compress`

Открытые атрибуты

- `ui32 cache`
- `const unsigned char * offset`

Объявления и описания членов структуры находятся в файле:

- `Thread_Pool+Archiver/quicklz.h`

3.7 Структура `qlz_hash_decompress`

Открытые атрибуты

- `const unsigned char * offset`

Объявления и описания членов структуры находятся в файле:

- `Thread_Pool+Archiver/quicklz.h`

3.8 Структура qlz_state_compress

Открытые атрибуты

- `size_t stream_counter`
- [qlz_hash_compress](#) hash [QLZ_HASH_VALUES]
- `unsigned char hash_counter [QLZ_HASH_VALUES]`

Объявления и описания членов структуры находятся в файле:

- Thread_Pool+Archiver/quicklz.h

3.9 Структура qlz_state_decompress

Открытые атрибуты

- [qlz_hash_decompress](#) hash [QLZ_HASH_VALUES]
- `unsigned char hash_counter [QLZ_HASH_VALUES]`
- `size_t stream_counter`

Объявления и описания членов структуры находятся в файле:

- Thread_Pool+Archiver/quicklz.h

3.10 Класс Tar

Производные классы: [lib_archiver](#) и [own_archiver](#).

Открытые члены

- `virtual int compress (const char *input, const char *output)=0`
- `virtual int decompress (const char *input, const char *output)=0`

Объявления и описания членов класса находятся в файле:

- Thread_Pool+Archiver/tar.h

3.11 Структура tarheader

Открытые атрибуты

- char filename [100]
- char filemode [8]
- char userid [8]
- char groupid [8]
- char filesize [12]
- char modiftime [12]
- char checksum [8]
- char filetype [1]
- char linkname [100]
- char indicator [6]
- char version [2]
- char username [32]
- char groupname [32]
- char majornumber [8]
- char minornumber [8]
- char prefix [155]
- char pad [12]

Объявления и описания членов структуры находятся в файле:

- Thread_Pool+Archiver/archive.h

3.12 Структура task

Открытые члены

- bool operator< (const task &other) const
- template<typename F, typename... Rest>
auto add_function (F &&f, Rest &&... rest) -> std::future< decltype(f(0, rest...))>
- int task_id ()
- void make_task (const char *name, int priority)
- int task_priority ()
- string task_name ()
- std::function< void(int id)> * task_func ()

Объявления и описания членов структуры находятся в файле:

- Thread_Pool+Archiver/TaskQueue.h

3.13 Класс TaskQueue

Открытые члены

- TaskQueue (int q, bool verbose)
- TaskQueue & operator= (const TaskQueue &rhs)
- void add_task (const char *name, int p, task obj)
- void push_to_end (std::thread::id this_id)
- void print (int mode, std::thread::id this_id)
- bool empty (std::thread::id this_id)
- void pop ()
- bool check_task_vector ()
- size_t size_function_queue ()
- int return_quesap ()
- void push (int k)
- int task_id ()
- std::function< void(int id)> * task_f ()
- string task_name ()

Объявления и описания членов классов находятся в файлах:

- Thread_Pool+Archiver/TaskQueue.h
- Thread_Pool+Archiver/TaskQueue.cpp

3.14 Класс Testing_class

Открытые члены

- string creativer ()
- bool isFilesEqual (const std::string &lFilePath, const std::string &rFilePath)
- bool exists (const char *name)
- double filesize (const char *filename)
- void creation_function (int p)
- bool extraction_function (int p)
- bool test_1 (int p)
- bool test_2 (int p)
- void create_file (int mode, vector< string > &filenames)
- bool sub_test_3 ()
- template<typename T >
string toString (T value)
- string merge (std::initializer_list< std::string > strList)
- template<typename ... Args>
bool test_3 (const Args &... args)
- bool test_4 (int w, int q)
- bool test_5 (const char *in, const char *out)
- bool test_gl (int w, int q)

Открытые атрибуты

- `vector< string > test_vector`
- `double size_after = 0`
- `double size_before = 0`

Объявления и описания членов классов находятся в файлах:

- `Thread_Pool+Archiver/tester.h`
- `Thread_Pool+Archiver/Tester.cpp`
- `Thread_Pool+Archiver/Tester_add.cpp`

3.15 Класс Thread_pool

Открытые члены

- `Thread_pool (int w, TaskQueue &obj, bool verbose)`
- `void work (TaskQueue &obj)`
- `bool run (TaskQueue &obj, std::thread::id this_id)`

Друзья

- `class TaskQueue`

Объявления и описания членов классов находятся в файлах:

- `Thread_Pool+Archiver/Thread_pool.h`
- `Thread_Pool+Archiver/Thread_pool.cpp`

Предметный указатель

bi_file, [2](#)

dictionary, [3](#)

lib_archiver, [3](#)

LZW_archiver, [3](#)

own_archiver, [4](#)

qlz_hash_compress, [4](#)

qlz_hash_decompress, [4](#)

qlz_state_compress, [5](#)

qlz_state_decompress, [5](#)

Tar, [5](#)

tarheader, [6](#)

task, [6](#)

TaskQueue, [7](#)

Testing_class, [7](#)

Thread_pool, [8](#)