

Шифр маршрутной перестановки

Создано системой Doxygen 1.8.13

Оглавление

1	Иерархический список классов	1
1.1	Иерархия классов	1
2	Алфавитный указатель классов	3
2.1	Классы	3
3	Список файлов	5
3.1	Файлы	5
4	Классы	7
4.1	Класс <code>cipher_error</code>	7
4.1.1	Подробное описание	8
4.2	Класс <code>SwapCipher</code>	8
4.2.1	Подробное описание	9
4.2.2	Методы	9
4.2.2.1	<code>decrypt()</code>	9
4.2.2.2	<code>encrypt()</code>	9
4.2.2.3	<code>getValidCipherText()</code>	10
4.2.2.4	<code>getValidKey()</code>	10
4.2.2.5	<code>getValidOpenText()</code>	11
5	Файлы	13
5.1	Файл <code>main.cpp</code>	13
5.1.1	Подробное описание	14
5.2	Файл <code>swarcipher.cpp</code>	14
5.2.1	Подробное описание	15
5.3	Файл <code>swarcipher.h</code>	15
5.3.1	Подробное описание	16
	Алфавитный указатель	17

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

invalid_argument	
cipher_error	7
SwapCipher	8

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	
Класс для обработки исключений	7
SwapCipher	
Шифрование методом табличной перестановки	8

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

main.cpp	Главный файл для модуля swarcipher	13
swarcipher.cpp	Исполняемый файл для модуля swarcipher	14
swarcipher.h	Заголовочный файл для модуля swarcipher	15

Глава 4

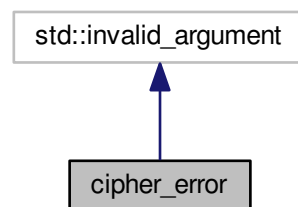
Классы

4.1 Класс `cipher_error`

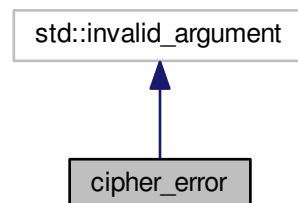
Класс для обработки исключений

```
#include <swarcipher.h>
```

Граф наследования: `cipher_error`:



Граф связей класса `cipher_error`:



Открытые члены

- `cipher_error (const std::string &what_arg)`
- `cipher_error (const char *what_arg)`

4.1.1 Подробное описание

Класс для обработки исключений

Состоит из двух методов. Для вывода исключения использует строку.

Объявления и описания членов класса находятся в файле:

- [swapcipher.h](#)

4.2 Класс SwapCipher

Шифрование методом табличной перестановки

```
#include <swapcipher.h>
```

Открытые члены

- [SwapCipher \(\)](#)=delete
конструктор без параметров запрещен
- [SwapCipher \(const int key1\)](#)
конструктор для установки ключа
- `std::string encrypt (const std::string &open_text)`
Зашифровывание
- `std::string decrypt (const std::string &cipher_text)`
Дешифровывание

Закрытые члены

- `void getValidKey (const int &key)`
Проверка ключа на валидность
- `std::string getValidOpenText (const std::string &s)`
Проверка шифруемого текста
- `std::string getValidCipherText (const std::string &s)`
Проверка дешифруемого текста

Закрытые данные

- `int key_stolbec`
ключ(количество столбцов матрицы)
- `int key_stroka`
количество строк матрицы
- `std::wstring real_cp`
строка передающаяся в decrypt.

4.2.1 Подробное описание

Шифрование методом табличной перестановки

Ключ устанавливается в конструкторе. Для зашифровывания и расшифровывания предназначены методы `encrypt` и `decrypt`.

Предупреждения

Реализация только для русского языка

4.2.2 Методы

4.2.2.1 `decrypt()`

```
std::string SwapCipher::decrypt (
    const std::string & cipher_text )
```

Дешифровывание

Аргументы

in	open_text	Закодированный текст. Не должен быть пустой строкой. Не должен быть единичной длины
----	-----------	---

Возвращает

Зашифрованная строка

Исключения

cipher_error ,если	текст пустой если текст еденичной длины если текст содержит пробелы
------------------------------------	---

4.2.2.2 `encrypt()`

```
std::string SwapCipher::encrypt (
    const std::string & open_text )
```

Зашифровывание

Аргументы

in	open_text	Открытый текст. Не должен быть пустой строкой. Не должен быть единичной длины Строчные символы автоматически преобразуются к прописным.
----	-----------	---

Возвращает

Зашифрованная строка

Исключения

cipher_error ,если	текст пустой если текст еденичной длины если текст содержит пробелы
------------------------------------	---

4.2.2.3 getValidCipherText()

```
std::string SwapCipher::getValidCipherText (
    const std::string & s ) [inline], [private]
```

Проверка дешифруемого текста

Проверки на все исключения "выбрасываемые" методом decrypt

Аргументы

in	cipher_text	Зашифрованный текст.
----	-------------	----------------------

Исключения

cipher_error ,если	текст пустой если текст еденичной длины если текст содержит пробелы
------------------------------------	---

4.2.2.4 getValidKey()

```
void SwapCipher::getValidKey (
    const int & key ) [private]
```

Проверка ключа на валидность

Проверки на все исключения "выбрасываемые" конструктором

Аргументы

in	key	Ключ.
----	-----	-------

Исключения

cipher_error	если ключ - меньше нуля или ноль.
------------------------------	-----------------------------------

4.2.2.5 `getValidOpenText()`

```
std::string SwapCipher::getValidOpenText (
    const std::string & s ) [inline], [private]
```

Проверка шифруемого текста

Проверки на все исключения "выбрасываемые" методом `encrypt`

Аргументы

in	open_text	Открытый текст.
----	-----------	-----------------

Исключения

cipher_error ,если	текст пустой если текст еденичной длины если текст содержит пробелы
------------------------------------	---

Объявления и описания членов классов находятся в файлах:

- [swapcipher.h](#)
- [swapcipher.cpp](#)

Глава 5

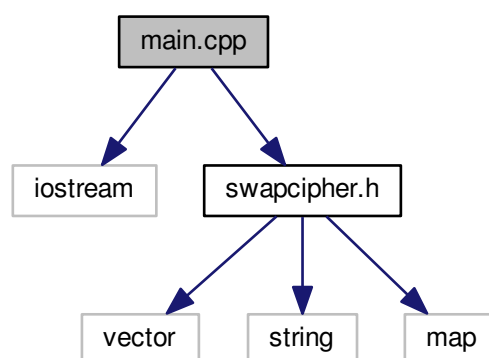
Файлы

5.1 Файл main.cpp

Главный файл для модуля swapcipher.

```
#include <iostream>
#include "swapcipher.h"
```

Граф включаемых заголовочных файлов для main.cpp:



Функции

- void check (const string &Text, const int key, const bool destructCipherText=false)
- int main (int argc, char **argv)

5.1.1 Подробное описание

Главный файл для модуля `swapcipher`.

Автор

Григорьев А.Д.

Версия

1.0

Дата

09.06.2019

Авторство

ИБСТ ПГУ

Предупреждения

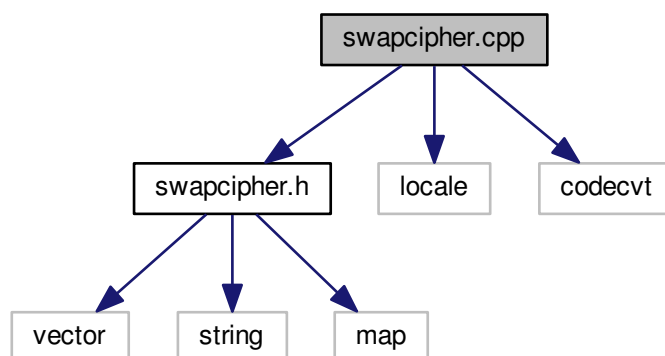
Это учебный пример

5.2 Файл `swapcipher.cpp`

Исполняемый файл для модуля `swapcipher`.

```
#include "swapcipher.h"  
#include <locale>  
#include <codecvt>
```

Граф включаемых заголовочных файлов для `swapcipher.cpp`:



Функции

- `std::locale loc ("ru_RU.UTF-8")`

Переменные

- `std::wstring_convert< std::codecvt_utf8< wchar_t >, wchar_t > codec`

5.2.1 Подробное описание

Исполняемый файл для модуля swarcipher.

Автор

Григорьев А.Д.

Версия

1.0

Дата

09.06.2019

Авторство

ИБСТ ПГУ

Предупреждения

Это учебный пример

5.3 Файл swarcipher.h

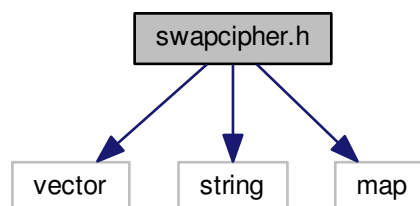
Заголовочный файл для модуля swarcipher.

```
#include <vector>
```

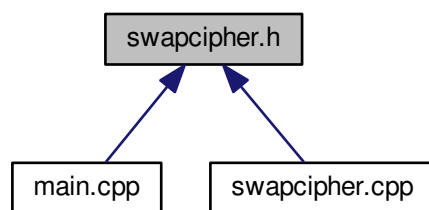
```
#include <string>
```

```
#include <map>
```

Граф включаемых заголовочных файлов для swarcipher.h:



Граф файлов, в которые включается этот файл:



Классы

- class [SwapCipher](#)
Шифрование методом табличной перестановки
- class [cipher_error](#)
Класс для обработки исключений

5.3.1 Подробное описание

Заголовочный файл для модуля `swarcipher`.

Автор

Григорьев А.Д.

Версия

1.0

Дата

09.06.2019

Авторство

ИБСТ ПГУ

Предупреждения

Это учебный пример

Предметный указатель

cipher_error, [7](#)

decrypt
 SwapCipher, [9](#)

encrypt
 SwapCipher, [9](#)

getValidCipherText
 SwapCipher, [10](#)

getValidKey
 SwapCipher, [10](#)

getValidOpenText
 SwapCipher, [10](#)

main.cpp, [13](#)

SwapCipher, [8](#)
 decrypt, [9](#)
 encrypt, [9](#)
 getValidCipherText, [10](#)
 getValidKey, [10](#)
 getValidOpenText, [10](#)

swapcipher.cpp, [14](#)

swapcipher.h, [15](#)