

Шифрование методом маршрутной перестановки

Создано системой Doxygen 1.9.4

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 cipher_error	7
4.1.1 Подробное описание	8
4.1.2 Конструктор(ы)	8
4.1.2.1 cipher_error() [1/2]	8
4.1.2.2 cipher_error() [2/2]	8
4.2 Класс Code	8
4.2.1 Подробное описание	9
4.2.2 Конструктор(ы)	9
4.2.2.1 Code()	9
4.2.3 Методы	10
4.2.3.1 decrypt()	10
4.2.3.2 encryption()	10
4.2.3.3 getValidCipherText()	10
4.2.3.4 getValidKey()	11
4.2.3.5 getValidOpenText()	11
5 Файлы	13
5.1 Файл modCipher.h	13
5.1.1 Подробное описание	14
5.2 modCipher.h	14
Предметный указатель	15

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

Code	8
std::invalid_argument	
cipher_error	7

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	Исключение, используемое для обозначения ошибок шифрования	7
Code	Класс для шифрования и дешифрования текста	8

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

modCipher.h	
Описание класса modCipher и связанных функций	13

Глава 4

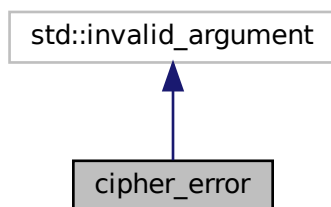
Классы

4.1 Класс `cipher_error`

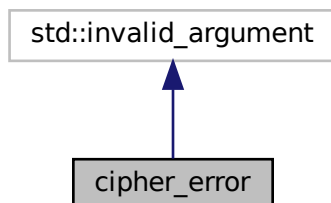
Исключение, используемое для обозначения ошибок шифрования.

```
#include <modCipher.h>
```

Граф наследования: `cipher_error`:



Граф связей класса `cipher_error`:



Открытые члены

- [cipher_error](#) (const std::string &what_arg)
Конструктор с сообщением об ошибке.
- [cipher_error](#) (const char *what_arg)
Конструктор с сообщением об ошибке.

4.1.1 Подробное описание

Исключение, используемое для обозначения ошибок шифрования.

4.1.2 Конструктор(ы)

4.1.2.1 cipher_error() [1/2]

```

cipher_error::cipher_error (
    const std::string & what_arg )  [inline], [explicit]

```

Конструктор с сообщением об ошибке.

Аргументы

what_arg	Описание ошибки.
----------	------------------

4.1.2.2 cipher_error() [2/2]

```

cipher_error::cipher_error (
    const char * what_arg )  [inline], [explicit]

```

Конструктор с сообщением об ошибке.

Аргументы

what_arg	Описание ошибки в виде C-строки.
----------	----------------------------------

Объявления и описания членов класса находятся в файле:

- [modCipher.h](#)

4.2 Класс Code

Класс для шифрования и дешифрования текста.

```
#include <modCipher.h>
```

Открытые члены

- `Code ()=delete`
Удалённый конструктор по умолчанию.
- `Code (int skey, const std::string &text)`
Конструктор с заданным ключом и текстом.
- `std::string encryption (const std::string &text)`
Шифрует текст.
- `std::string decrypt (const std::string &text, const std::string &open_text)`
Дешифрует текст.

Закрытые члены

- `int getValidKey (int key, const std::string &text)`
Проверяет и возвращает корректный ключ.
- `std::string getValidOpenText (const std::string &ts)`
Проверяет и возвращает корректный открытый текст.
- `std::string getValidCipherText (const std::string &s, const std::string &open_text)`
Проверяет и возвращает корректный зашифрованный текст.

Закрытые данные

- `int key`
Ключ для шифрования

4.2.1 Подробное описание

Класс для шифрования и дешифрования текста.

4.2.2 Конструктор(ы)

4.2.2.1 Code()

```
Code::Code (
    int skey,
    const std::string & text )
```

Конструктор с заданным ключом и текстом.

Аргументы

skey	Ключ шифрования.
text	Текст для шифрования.

4.2.3 Методы

4.2.3.1 decrypt()

```
std::string Code::decrypt (
    const std::string & text,
    const std::string & open_text )
```

Дешифрует текст.

Аргументы

text	Зашифрованный текст.
open_text	Открытый текст.

Возвращает

Дешифрованный текст.

4.2.3.2 encryption()

```
std::string Code::encryption (
    const std::string & text )
```

Шифрует текст.

Аргументы

text	Текст для шифрования.
------	-----------------------

Возвращает

Зашифрованный текст.

4.2.3.3 getValidCipherText()

```
std::string Code::getValidCipherText (
    const std::string & s,
    const std::string & open_text ) [private]
```

Проверяет и возвращает корректный зашифрованный текст.

Аргументы

s	Исходная строка.
open_text	Открытый текст.

Возвращает

Корректированный зашифрованный текст.

4.2.3.4 getValidKey()

```
int Code::getValidKey (
    int key,
    const std::string & text ) [private]
```

Проверяет и возвращает корректный ключ.

Аргументы

key	Исходный ключ.
text	Текст для шифрования или дешифрования.

Возвращает

Корректированный ключ.

4.2.3.5 getValidOpenText()

```
std::string Code::getValidOpenText (
    const std::string & s ) [private]
```

Проверяет и возвращает корректный открытый текст.

Аргументы

s	Исходная строка.
---	------------------

Возвращает

Корректированный открытый текст.

Объявления и описания членов класса находятся в файле:

- [modCipher.h](#)

Глава 5

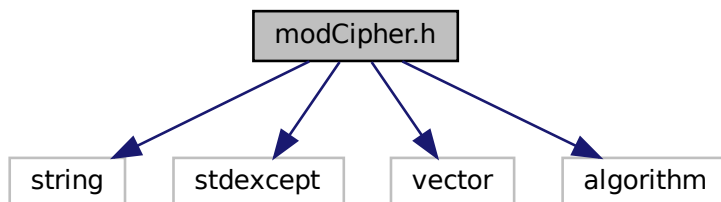
Файлы

5.1 Файл modCipher.h

Описание класса modCipher и связанных функций.

```
#include <string>
#include <stdexcept>
#include <vector>
#include <algorithm>
```

Граф включаемых заголовочных файлов для modCipher.h:



Классы

- class [cipher_error](#)
Исключение, используемое для обозначения ошибок шифрования.
- class [Code](#)
Класс для шифрования и дешифрования текста.

5.1.1 Подробное описание

Описание класса modCipher и связанных функций.

Автор

- Беликов А.М.

Версия

- 1.0

Дата

- 02.12.2024

-

5.2 modCipher.h

[См. документацию.](#)

```
1
7 #pragma once
8 #include <string>
9 #include <stdexcept>
10 #include <vector>
11 #include <algorithm>
12
17 class cipher_error : public std::invalid_argument {
18 public:
23     explicit cipher_error(const std::string& what_arg) : std::invalid_argument(what_arg) {}
24
29     explicit cipher_error(const char* what_arg) : std::invalid_argument(what_arg) {}
30 };
31
36 class Code {
37 private:
38     int key;
39
46     int getValidKey(int key, const std::string& text);
47
53     std::string getValidOpenText(const std::string& s);
54
61     std::string getValidCipherText(const std::string& s, const std::string& open_text);
62
63 public:
67     Code() = delete;
68
74     Code(int skey, const std::string& text);
75
81     std::string encryption(const std::string& text);
82
89     std::string decrypt(const std::string& text, const std::string& open_text);
90 };
```

Предметный указатель

- cipher_error, [7](#)
 - cipher_error, [8](#)
- Code, [8](#)
 - Code, [9](#)
 - decrypt, [10](#)
 - encryption, [10](#)
 - getValidCipherText, [10](#)
 - getValidKey, [11](#)
 - getValidOpenText, [11](#)
- decrypt
 - Code, [10](#)
- encryption
 - Code, [10](#)
- getValidCipherText
 - Code, [10](#)
- getValidKey
 - Code, [11](#)
- getValidOpenText
 - Code, [11](#)
- modCipher.h, [13](#)