

RouteTableCipher

1.0

Создано системой Doxygen 1.8.17

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
4.1.3 Методы	9
4.1.3.1 getValidKey()	9
4.1.3.2 getValidText()	9
4.2 Класс RouteTableCipher	9
4.2.1 Подробное описание	10
4.2.2 Конструктор(ы)	10
4.2.2.1 RouteTableCipher()	10
4.2.3 Методы	10
4.2.3.1 decrypt()	11
4.2.3.2 encrypt()	11
5 Файлы	13
5.1 Файл RouteTableCipher.h	13
5.1.1 Подробное описание	14
Предметный указатель	15

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

invalid_argument	
cipher_error	7
RouteTableCipher	9

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	Данный класс нужен для обработки ошибок, которые могут возникнуть в течение работы программы	7
RouteTableCipher	Данный класс реализует шифрование сообщений методом табличной маршрутной перестановки	9

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

[RouteTableCipher.h](#)

Шифрование методом табличной маршрутной перестановки 13

Глава 4

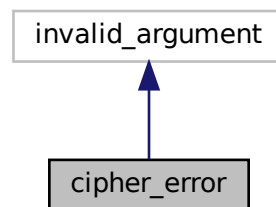
Классы

4.1 Класс cipher_error

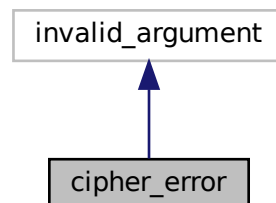
Данный класс нужен для обработки ошибок, которые могут возникнуть в течение работы программы

```
#include <RouteTableCipher.h>
```

Граф наследования: cipher_error:



Граф связей класса cipher_error:



Открытые члены

- `cipher_error` (`const string &what_arg`)
Конструктор принимает на вход строку и возбуждает исключение
- `cipher_error` (`const char *what_arg`)
Конструктор принимает на вход си-строку и возбуждает исключение

Открытые статические члены

- static int `getValidKey` (`const string skey`)
Метод, который реализует проверку на валидацию ключа
- static int `getValidText` (`const string text`)
Метод, который реализует проверку на валидацию зашифрованного и расшифрованного текста

4.1.1 Подробное описание

Данный класс нужен для обработки ошибок, которые могут возникнуть в течение работы программы

4.1.2 Конструктор(ы)

4.1.2.1 `cipher_error()` [1/2]

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

Конструктор принимает на вход строку и возбуждает исключение

Аргументы

what_arg	
----------	--

4.1.2.2 `cipher_error()` [2/2]

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

Конструктор принимает на вход си-строку и возбуждает исключение

Аргументы

what_arg	
----------	--

4.1.3 Методы

4.1.3.1 getValidKey()

```
int cipher_error::getValidKey (
    const string skey ) [static]
```

Метод, который реализует проверку на валидацию ключа

Аргументы

skey	
------	--

Исключения

cipher_error ,если	в ключе содержатся недопустимые символы или он является пустым
------------------------------------	--

4.1.3.2 getValidText()

```
int cipher_error::getValidText (
    const string text ) [static]
```

Метод, который реализует проверку на валидацию зашифрованного и расшифрованного текста

Аргументы

text	
------	--

Исключения

cipher_error ,если	в тексте присутствуют русские буквы или он является пустым
------------------------------------	--

Объявления и описания членов классов находятся в файлах:

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

4.2 Класс RouteTableCipher

Данный класс реализует шифрование сообщений методом табличной маршрутной перестановки

```
#include <RouteTableCipher.h>
```

Открытые члены

- `RouteTableCipher ()=delete`
Запрет конструктора без параметров
- `RouteTableCipher (const int &skey)`
Конструктор, выполняющий установку ключа
- `string encrypt (string open_text, const int &skey)`
Метод, предназначенный для шифрования сообщений методом маршрутной табличной перестановки
- `string decrypt (string cipher_text, const int &skey)`
Метод, предназначенный для расшифрования сообщений

Закрытые данные

- `int key`
Атрибут класса, хранящий ключ шифрования и расшифрования

4.2.1 Подробное описание

Данный класс реализует шифрование сообщений методом табличной маршрутной перестановки

Предупреждения

Реализация производится с использованием английского языка

4.2.2 Конструктор(ы)

4.2.2.1 RouteTableCipher()

```
RouteTableCipher::RouteTableCipher (
    const int & skey )
```

Конструктор, выполняющий установку ключа

Аргументы

skey	- ключ в виде целого числа
------	----------------------------

Внутри конструктора атрибуту класса присваивается значение входного параметра

4.2.3 Методы

4.2.3.1 decrypt()

```
string RouteTableCipher::decrypt (
    string cipher_text,
    const int & skey )
```

Метод, предназначенный для расшифрования сообщений

Аргументы

cipher_text	- зашифрованный текст
skey	- ключ

Метод формирует таблицу, записывает в каждую ячейку символ зашифрованного сообщения и формирует расшифрованное сообщение.

Возвращает

Метод возвращает расшифрованное сообщение типа string

4.2.3.2 encrypt()

```
string RouteTableCipher::encrypt (
    string open_text,
    const int & skey )
```

Метод, предназначенный для шифрования сообщений методом маршрутной табличной перестановки

Аргументы

open_text	- открытый текст
skey	- ключ

Метод формирует таблицу, в каждую ячейку которой записывается символ введённого пользователем сообщения;

Таблица состоит из количества столбцов, введённого пользователем, количество строк определяется в зависимости от длины сообщения;

Метод по маршруту записи фиксирует текст в таблице и по маршруту считывания формирует зашифрованное сообщение.

Возвращает

Метод возвращает зашифрованное сообщение типа string

Объявления и описания членов классов находятся в файлах:

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

Глава 5

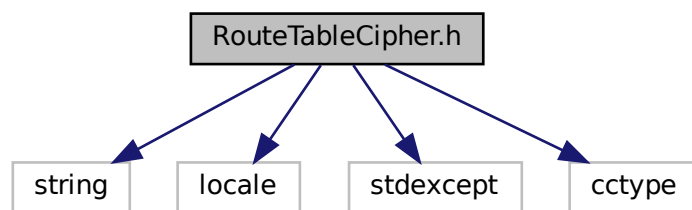
Файлы

5.1 Файл RouteTableCipher.h

Шифрование методом табличной маршрутной перестановки

```
#include <string>
#include <locale>
#include <stdexcept>
#include <cctype>
```

Граф включаемых заголовочных файлов для RouteTableCipher.h:



Классы

- class [RouteTableCipher](#)

Данный класс реализует шифрование сообщений методом табличной маршрутной перестановки

- class [cipher_error](#)

Данный класс нужен для обработки ошибок, которые могут возникнуть в течение работы программы

5.1.1 Подробное описание

Шифрование методом табличной маршрутной перестановки

Автор

Еремян М.Т.

Дата

20/04/22

Версия

1.0

Авторство

ИБСТ ПГУ

Предметный указатель

- cipher_error, [7](#)
 - cipher_error, [8](#)
 - getValidKey, [9](#)
 - getValidText, [9](#)
- decrypt
 - RouteTableCipher, [10](#)
- encrypt
 - RouteTableCipher, [11](#)
- getValidKey
 - cipher_error, [9](#)
- getValidText
 - cipher_error, [9](#)
- RouteTableCipher, [9](#)
 - decrypt, [10](#)
 - encrypt, [11](#)
 - RouteTableCipher, [10](#)
- RouteTableCipher.h, [13](#)