

Шифр маршрутной перестановки

1.0

Создано системой Doxygen 1.9.1

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 TableCipher	9
4.2.1 Подробное описание	9
4.2.2 Конструктор(ы)	9
4.2.2.1 TableCipher()	9
4.2.3 Методы	10
4.2.3.1 decrypt()	10
4.2.3.2 encrypt()	10
4.2.3.3 getValidKey()	11
4.2.3.4 getValidText()	11
5 Файлы	13
5.1 Файл TableCipher.h	13
5.1.1 Подробное описание	14
Предметный указатель	15

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

std::invalid_argument	
cipher_error	7
TableCipher	9

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	
Обработка исключений	7
TableCipher	
Шифрование методом маршрутной перестановки	9

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

[TableCipher.h](#)

Заголовочный файл для модуля маршрутной перестановки 13

Глава 4

Классы

4.1 Класс `cipher_error`

Обработка исключений.

```
#include <TableCipher.h>
```

Граф наследования: `cipher_error`:



Граф связей класса `cipher_error`:



Открытые члены

- `cipher_error` (`const std::string &what_arg`)
Конструктор с параметром типа `const std::string`.
- `cipher_error` (`const char *what_arg`)
Конструктор с параметром типа `const char`.

4.1.1 Подробное описание

Обработка исключений.

Класс `cipher_error` наследуется от `std::invalid_argument` и используется для обработки ошибок, возникающих при некорректных входных данных (например, невалидный ключ или текст).

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 параметром типа `const std::string`.

Создает объект исключения с указанным сообщением об ошибке.

Аргументы

<code>what_arg</code>	Сообщение об ошибке.
-----------------------	----------------------

4.1.2.2 `cipher_error()` [2/2]

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

Конструктор с параметром типа `const char`.

Создает объект исключения с указанным сообщением об ошибке.

Аргументы

<code>what_arg</code>	Сообщение об ошибке.
-----------------------	----------------------

Объявления и описания членов класса находятся в файле:

- [TableCipher.h](#)

4.2 Класс TableCipher

Шифрование методом маршрутной перестановки

```
#include <TableCipher.h>
```

Открытые члены

- [TableCipher](#) (const int &key)
Конструктор с параметрами
- std::wstring [encrypt](#) (const std::wstring &encrypt_text)
Метод для шифрования текста.
- std::wstring [decrypt](#) (const std::wstring &decrypt_text)
Метод для расшифровки текста.

Закрытые члены

- int [getValidKey](#) (const int &key)
Проверка ключа на валидность.
- std::wstring [getValidText](#) (const std::wstring &text)
Проверка текста на валидность.

Закрытые данные

- int key1

4.2.1 Подробное описание

Шифрование методом маршрутной перестановки

Класс [TableCipher](#) предоставляет методы для зашифрования и расшифрования текста с использованием табличного шифра. Ключ для зашифрования и расшифрования должен быть положительным целым числом.

4.2.2 Конструктор(ы)

4.2.2.1 TableCipher()

```
TableCipher::TableCipher (  
    const int & key )
```

Конструктор с параметрами

Создает объект [TableCipher](#) с указанным ключом.

Аргументы

key	Ключ для шифрования и дешифрования.
-----	-------------------------------------

Исключения

cipher_error	Если ключ не является положительным целым числом.
------------------------------	---

4.2.3 Методы

4.2.3.1 decrypt()

```
std::wstring TableCipher::decrypt (  
    const std::wstring & decrypt_text )
```

Метод для расшифровки текста.

Расшифровывает текст с использованием табличного шифра и ключа, указанного при создании объекта.

Аргументы

decrypt_text	Текст, который нужно расшифровать.
--------------	------------------------------------

Возвращает

Расшифрованный текст.

Исключения

cipher_error	Если текст содержит недопустимые символы.
------------------------------	---

4.2.3.2 encrypt()

```
std::wstring TableCipher::encrypt (  
    const std::wstring & encrypt_text )
```

Метод для шифрования текста.

Шифрует текст с использованием табличного шифра и ключа, указанного при создании объекта.

Аргументы

encrypt_text	Текст, который нужно зашифровать.
--------------	-----------------------------------

Возвращает

Зашифрованный текст.

Исключения

cipher_error	Если текст содержит недопустимые символы.
------------------------------	---

4.2.3.3 getValidKey()

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

Проверка ключа на валидность.

Метод проверяет, является ли ключ положительным целым числом.

Аргументы

key	Ключ, который нужно проверить.
-----	--------------------------------

Возвращает

Валидный ключ.

Исключения

cipher_error	Если ключ не является положительным целым числом.
------------------------------	---

4.2.3.4 getValidText()

```
std::wstring TableCipher::getValidText (  
    const std::wstring & text ) [private]
```

Проверка текста на валидность.

Метод проверяет, содержит ли текст только буквы алфавита.

Аргументы

text	Текст, который нужно проверить.
------	---------------------------------

Возвращает

Валидный текст.

Исключения

cipher_error	Если текст содержит недопустимые символы.
------------------------------	---

Объявления и описания членов классов находятся в файлах:

- [TableCipher.h](#)
- [tableCipher.cpp](#)

Глава 5

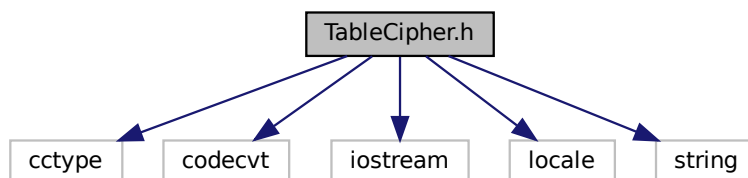
Файлы

5.1 Файл TableCipher.h

Заголовочный файл для модуля маршрутной перестановки

```
#include <cctype>
#include <codecvt>
#include <iostream>
#include <locale>
#include <string>
```

Граф включаемых заголовочных файлов для TableCipher.h:



Классы

- class [TableCipher](#)
Шифрование методом маршрутной перестановки
- class [cipher_error](#)
Обработка исключений.

5.1.1 Подробное описание

Заголовочный файл для модуля маршрутной перестановки

Автор

Синельникова Т.А.

Версия

1.0

Дата

01.12.2024

Авторство

ИБСТ ПГУ

Предупреждения

Это учебный пример

Предметный указатель

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