

tableCipherRUS

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 tableCipher | 7 |
| 4.1.1 Подробное описание | 7 |
| 4.1.2 Конструктор(ы) | 7 |
| 4.1.2.1 tableCipher() | 7 |
| 4.1.3 Методы | 8 |
| 4.1.3.1 decrypt() | 8 |
| 4.1.3.2 encrypt() | 8 |
| 4.1.3.3 getValidKey() | 9 |
| 4.2 Класс tableCipher_error | 9 |
| 4.2.1 Подробное описание | 10 |
| 5 Файлы | 11 |
| 5.1 Файл table.h | 11 |
| 5.1.1 Подробное описание | 12 |
| 5.2 table.h | 12 |
| Предметный указатель | 13 |

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

| | |
|-----------------------------|---|
| std::invalid_argument | |
| tableCipher_error | 9 |
| tableCipher | 7 |

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

| | |
|------------------------------------------------------|---|
| tableCipher | |
| Шифрование методом табличной маршрутной перестановки | 7 |
| tableCipher_error | |
| Класс исключений | 9 |

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

| | | |
|-------------------------|--------------------------------------------------|----|
| table.h | Заголовочный файл для модуля table.cpp | 11 |
|-------------------------|--------------------------------------------------|----|

Глава 4

Классы

4.1 Класс tableCipher

Шифрование методом табличной маршрутной перестановки

```
#include <table.h>
```

Открытые члены

- `tableCipher` (unsigned int `columns`)
Конструктор для установки ключа
- `std::wstring encrypt` (const `std::wstring` &`sourcetext`)
Зашифровывание
- `std::wstring decrypt` (const `std::wstring` &`ciphertext`)
Расшифровывание
- unsigned int `getValidKey` (unsigned int `key`)
Метод проверки ключа на валидность

Закрытые данные

- unsigned int `columns`
Ключ=количество столбцов

4.1.1 Подробное описание

Шифрование методом табличной маршрутной перестановки

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

4.1.2 Конструктор(ы)

4.1.2.1 `tableCipher()`

```
tableCipher::tableCipher (  
    unsigned int columns )
```

Конструктор для установки ключа

Аргументы

| | | |
|----|---------|-------------|
| in | columns | Строка-ключ |
|----|---------|-------------|

4.1.3 Методы

4.1.3.1 decrypt()

```
std::wstring tableCipher::decrypt (
    const std::wstring & ciphertext )
```

Расшифровывание

Аргументы

| | | |
|----|------------|-----------------------------------------------------|
| in | ciphertext | Зашифрованный текст. Не должен быть пустой строкой. |
|----|------------|-----------------------------------------------------|

Возвращает

Расшифрованная строка

Исключения

| | |
|-----------------------------------------|--------------|
| tableCipher_error ,если | текст пустой |
|-----------------------------------------|--------------|

4.1.3.2 encrypt()

```
std::wstring tableCipher::encrypt (
    const std::wstring & sourcetext )
```

Зашифровывание

Аргументы

| | | |
|----|------------|------------------------------------------------|
| in | sourcetext | Исходный текст. Не должен быть пустой строкой. |
|----|------------|------------------------------------------------|

Возвращает

Зашифрованная строка

Исключения

| | |
|-----------------------------------------|--------------|
| tableCipher_error ,если | текст пустой |
|-----------------------------------------|--------------|

4.1.3.3 getValidKey()

```
unsigned int tableCipher::getValidKey (
    unsigned int key )
```

Метод проверки ключа на валидность

Аргументы

| | | |
|----|---|------|
| in | s | Ключ |
|----|---|------|

Возвращает

s Валидный ключ

Исключения

| | |
|-----------------------------------------|-----------------------------------------------------------------------|
| tableCipher_error ,если | ключ пустой, слишком простой(равен 1) или слишком большой(больше 100) |
|-----------------------------------------|-----------------------------------------------------------------------|

Объявления и описания членов классов находятся в файлах:

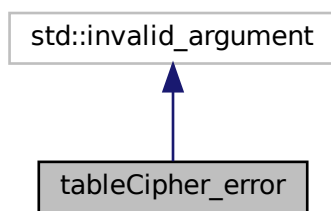
- [table.h](#)
- [table.cpp](#)

4.2 Класс tableCipher_error

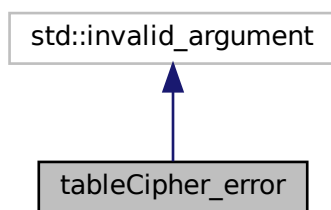
Класс исключений

```
#include <table.h>
```

Граф наследования:tableCipher_error:



Граф связей класса tableCipher_error:



Открытые члены

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

4.2.1 Подробное описание

Класс исключений

Класс отлова исключений

Объявления и описания членов класса находятся в файле:

- [table.h](#)

Глава 5

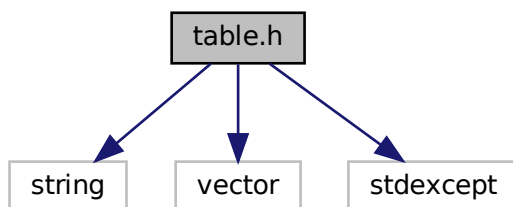
Файлы

5.1 Файл table.h

Заголовочный файл для модуля table.cpp.

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

Граф включаемых заголовочных файлов для table.h:



Классы

- class `tableCipher`
Шифрование методом табличной маршрутной перестановки
- class `tableCipher_error`
Класс исключений

5.1.1 Подробное описание

Заголовочный файл для модуля table.cpp.

Автор

Храбров А.А.

Версия

1.0

Дата

09.12.2024

Авторство

ИБСТ ПГУ

Предупреждения

Тестовый

5.2 table.h

[См. документацию.](#)

```
1
10 #pragma once
11 #include <string>
12 #include <vector>
13 #include <stdexcept>
14
22 class tableCipher {
23 public:
27     tableCipher(unsigned int columns);
28
35     std::wstring encrypt(const std::wstring& sourcetext);
36
43     std::wstring decrypt(const std::wstring& ciphertext);
44
52     unsigned int getValidKey(unsigned int key);
53
54 private:
55     unsigned int columns;
56 };
57
62 class tableCipher_error: public std::invalid_argument
63 {
64 public:
65     explicit tableCipher_error(const std::string& what_arg):
66         std::invalid_argument(what_arg) {}
67     explicit tableCipher_error(const char* what_arg):
68         std::invalid_argument(what_arg) {}
69 };
```


Предметный указатель

- decrypt
 - tableCipher, [8](#)
- encrypt
 - tableCipher, [8](#)
- getValidKey
 - tableCipher, [9](#)
- table.h, [11](#)
- tableCipher, [7](#)
 - decrypt, [8](#)
 - encrypt, [8](#)
 - getValidKey, [9](#)
 - tableCipher, [7](#)
- tableCipher_error, [9](#)