

Шифрование методом табличной перестановки.

1.0

Создано системой Doxygen 1.8.17

1 Алфавитный указатель классов	1
1.1 Классы	1
2 Список файлов	1
2.1 Файлы	1
3 Классы	1
3.1 Класс <code>Tabl_Marsh</code>	1
3.1.1 Подробное описание	2
3.1.2 Конструктор(ы)	2
3.1.3 Методы	2
4 Файлы	4
4.1 Файл <code>Tabl_Marsh.h</code>	4
4.1.1 Подробное описание	4
Предметный указатель	5

1 Алфавитный указатель классов

1.1 Классы

Классы с их кратким описанием.

<code>Tabl_Marsh</code>	
Класс зашифрования и расшифрования текста алгоритмом табличной перестановки	1

2 Список файлов

2.1 Файлы

Полный список документированных файлов.

<code>Tabl_Marsh.h</code>	
Описание класса <code>Tabl_Marsh</code>	4

3 Классы

3.1 Класс `Tabl_Marsh`

Класс зашифрования и расшифрования текста алгоритмом табличной перестановки.

```
#include <Tabl_Marsh.h>
```

Открытые члены

- `Tabl_Marsh ()`
Конструктор по умолчанию.
- `Tabl_Marsh (int &skey)`
Конструктор с установкой ключа.
- `wstring encrypt (const wstring &open_text)`
Метод зашифрования табличной перестановкой.
- `wstring decrypt (const wstring &cipher_text)`
Метод расшифрования текста, зашифрованного табличной перестановкой.
- `void Check (int skey)`
Проверка ключа.

Закрытые данные

- `int key`
Ключ, отвечающий за количество столбцов в таблице.
- `wstring EngAlph =L"ABCDEFGHIJKLMNOPQRSTUVWXYZ"`
Алфавит для английских символов.
- `wstring RusAlph =L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"`
Алфавит для русских символов.

3.1.1 Подробное описание

Класс зашифрования и расшифрования текста алгоритмом табличной перестановки.

3.1.2 Конструктор(ы)

3.1.2.1 `Tabl_Marsh()` `Tabl_Marsh::Tabl_Marsh (int &skey)`

Конструктор с установкой ключа.

Аргументы

<code>skey</code>	- ключ, вводимый пользователем.
-------------------	---------------------------------

3.1.3 Методы

3.1.3.1 `Check()` `void Tabl_Marsh::Check (int skey)`

Проверка ключа.

Аргументы

skey	- ключ, подлежащий проверке.
------	------------------------------

Данный метод проверяет правильность ввода ключа. Исключения выбрасываются если: был введен символ не являющийся цифрой, число не является положительным целочисленным.

3.1.3.2 `decrypt()` `wstring Tabl_Marsh::decrypt (`
`const wstring & cipher_text)`

Метод расшифрования текста, зашифрованного табличной перестановкой.

Аргументы

cipher_text	- текст, который необходимо расшифровать.
-------------	---

Полученный на вход текст, проверяется на пустоту, если он не проходит проверку, то выбрасывается исключение. На основе ключа создается таблица, в которую будут записываться символы текста. После записи текста в таблицу, эти же символы выписываются по правилу табличной перестановки для расшифрования.

Возвращает

Расшифрованный текст.

3.1.3.3 `encrypt()` `wstring Tabl_Marsh::encrypt (`
`const wstring & open_text)`

Метод зашифрования табличной перестановкой.

Аргументы

open_text	- текст, который необходимо зашифровать.
-----------	--

Полученный на вход текст, проверяется на пустоту, если он не проходит проверку, то выбрасывается исключение. На основе ключа создается таблица, в которую будут записываться символы текста. После записи текста в таблицу, эти же символы выписываются по правилу табличной перестановки.

Возвращает

Текст зашифрованный методом табличной перестановки.

Объявления и описания членов класса находятся в файле:

- [Tabl_Marsh.h](#)

4 Файлы

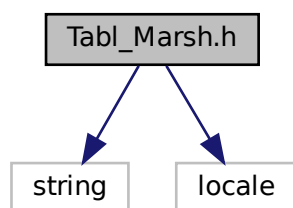
4.1 Файл Tabl_Marsh.h

Описание класса [Tabl_Marsh](#).

```
#include <string>
```

```
#include <locale>
```

Граф включаемых заголовочных файлов для Tabl_Marsh.h:



Классы

- class [Tabl_Marsh](#)

Класс зашифрования и расшифрования текста алгоритмом табличной перестановки.

4.1.1 Подробное описание

Описание класса [Tabl_Marsh](#).

Автор

Рясков.С.С.

Версия

1.0

Дата

29.05.2021

Авторство

ИБСТ ПГУ

Предметный указатель

Check

 Tabl_Marsh, [2](#)

decrypt

 Tabl_Marsh, [3](#)

encrypt

 Tabl_Marsh, [3](#)

Tabl_Marsh, [1](#)

 Check, [2](#)

 decrypt, [3](#)

 encrypt, [3](#)

 Tabl_Marsh, [2](#)

Tabl_Marsh.h, [4](#)