

PermutationCipher

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
4.2 Класс PermutationCipher	8
4.2.1 Подробное описание	8
4.2.2 Конструктор(ы)	8
4.2.2.1 PermutationCipher()	8
4.2.3 Методы	9
4.2.3.1 CoderPermutationCipher()	9
4.2.3.2 DecoderPermutationCipher()	10
5 Файлы	11
5.1 Файл PermutationCipher.h	11
5.1.1 Подробное описание	12
Предметный указатель	13

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

invalid_argument	
cipher_error	7
PermutationCipher	8

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	7
PermutationCipher	
Класс для расшифрования и шифрования шифра маршрутной табличной перестановки	8

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

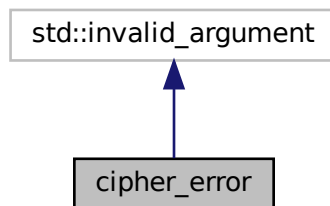
PermutationCipher.h	
Описание класса PermutationCipher	11

Глава 4

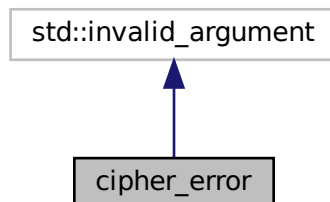
Классы

4.1 Класс `cipher_error`

Граф наследования: `cipher_error`:



Граф связей класса `cipher_error`:



Открытые члены

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

4.1.1 Подробное описание

предназначенный для возбуждения исключений.

Объявления и описания членов класса находятся в файле:

- [PermutationCipher.h](#)

4.2 Класс PermutationCipher

Класс для расшифрования и шифрования шифра маршрутной табличной перестановки.

```
#include <PermutationCipher.h>
```

Открытые члены

- [PermutationCipher](#) ()=delete
Конструктор без параметров.
- [PermutationCipher](#) (int w)
Конструктор для ключа
- wstring [CoderPermutationCipher](#) ([PermutationCipher](#) w, wstring &s)
Метод , который предназначен для шифрования шифром табличной маршрутной перестановки
- wstring [DecoderPermutationCipher](#) ([PermutationCipher](#) w, wstring &s)
Метод , который предназначен для расшифрования шифром табличной маршрутной перестановки

Закрытые члены

- wstring getValidOpenText (const std::wstring &s)
- wstring getValidCipherText (const std::wstring &s)
- int getValidKey (const int k, const std::wstring &s)

Закрытые данные

- int k
Атрибут, который хранит ключ для расшифрования или шифрования.

4.2.1 Подробное описание

Класс для расшифрования и шифрования шифра маршрутной табличной перестановки.

4.2.2 Конструктор(ы)

4.2.2.1 PermutationCipher()

```
PermutationCipher::PermutationCipher (  
    int w )
```

Конструктор для ключа

Аргументы

целочисленное	значение - ключ
---------------	-----------------

значение, приходящее на вход записывается в "private" в атрибут с названием "k"

4.2.3 Методы

4.2.3.1 CoderPermutationCipher()

```
wstring PermutationCipher::CoderPermutationCipher (
    PermutationCipher w,
    wstring & s )
```

Метод , который предназначен для шифрования шифром табличной маршрутной перестановки

Аргументы

экземляр	класса "PermutationCipher", в котором устанавливается ключ
wstring	- строка, которую нужно зашифровать

Сначала вычисляется размер таблицы.

```
if (s.size()%w.k!=0) {
    h=s.size()/w.k+1;
} else {
    h=s.size()/w.k;
} // Вычисление размера таблицы.
```

После создаётся двумерный массив "wchar_t" Кол-во строк вычисляется по формуле, а количество столбцов - это ключ, который устанавливается в классе "PermutationCipher".

```
wchar_t a[h][w.k];
```

Затем в созданный двумерный массив записываются символы строки, которую нужно зашифровать.

Запись происходит по строкам.

```
for (int i=0; i<h; i++) {
    for (int j=0; j<w.k; j++) {
        if (k<s.size()) {
            a[i][j]=s[k];
            k++;
        } else a[i][j]=' ';
    }
}
```

Далее идёт сам процесс шифрования.

```
for (int i=0; i<w.k; i++) {
    for (int j=0; j<h; j++) {
        code+=a[j][i];
    }
}
```

Возвращает

"code" - зашифрованная строка типа "wstring"

4.2.3.2 DecoderPermutationCipher()

```
wstring PermutationCipher::DecoderPermutationCipher (
    PermutationCipher w,
    wstring & s )
```

Метод , который предназначен для расшифрования шифром табличной маршрутной перестановки

Аргументы

экземляр	класса " PermutationCipher ", в котором устанавливается ключ
----------	--

Сначала вычисляется размер таблицы.

```
if (s.size()%w.k!=0) {
    h=s.size()/w.k+1;
} else {
    h=s.size()/w.k;
}
```

После создаётся двумерный массив "`wchar_t`" Кол-во строк вычисляется по формуле, а количество столбцов - это ключ, который устанавливается в классе "[PermutationCipher](#)".

```
wchar_t a[h][w.k];
```

Затем в созданный двумерный массив записываются символы строки, которую нужно расшифровать. Запись происходит по строкам.

```
for (int i=0; i<w.k; i++) {
    for (int j=0; j<h; j++) {
        a[j][i]=s[k];
        k++;
    }
}
```

Аргументы

wstring	- строка, которую нужно расшифровать
---------	--------------------------------------

Далее идёт сам процесс шифрования.

```
for (int i=0; i<h; i++) {
    for (int j=0; j<w.k; j++) {
        decode+=a[i][j];
    }
}
```

Возвращает

"decode" - зашифрованная строка типа "wstring"

Объявления и описания членов классов находятся в файлах:

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

Глава 5

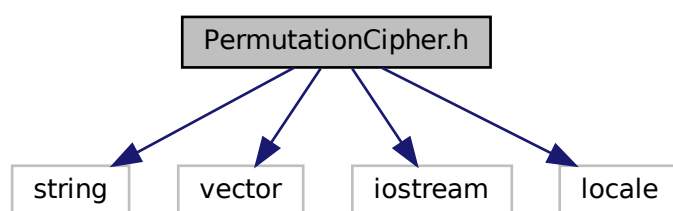
Файлы

5.1 Файл PermutationCipher.h

Описание класса [PermutationCipher](#).

```
#include <string>
#include <vector>
#include <iostream>
#include <locale>
```

Граф включаемых заголовочных файлов для PermutationCipher.h:



Классы

- class [PermutationCipher](#)

Класс для расшифрования и шифрования шифра маршрутной табличной перестановки.

- class [cipher_error](#)

5.1.1 Подробное описание

Описание класса [PermutationCipher](#).

Автор

Корзин А.А.

Версия

1.0

Дата

02.06.2021

Авторство

ИБСТ ПГУ

Предметный указатель

cipher_error, [7](#)

CoderPermutationCipher
 PermutationCipher, [9](#)

DecoderPermutationCipher
 PermutationCipher, [9](#)

PermutationCipher, [8](#)
 CoderPermutationCipher, [9](#)
 DecoderPermutationCipher, [9](#)
 PermutationCipher, [8](#)
PermutationCipher.h, [11](#)