

modAlphaCipher

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 modAlphaCipher . . . . .	8
4.2.1 Подробное описание . . . . .	9
4.2.2 Конструктор(ы) . . . . .	9
4.2.2.1 modAlphaCipher() . . . . .	9
4.2.3 Методы . . . . .	9
4.2.3.1 convert() [1/2] . . . . .	9
4.2.3.2 convert() [2/2] . . . . .	10
5 Файлы	11
5.1 Файл modAlphaCipher.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
modAlphaCipher . . . . .	8



## Глава 2

# Алфавитный указатель классов

### 2.1 Классы

Классы с их кратким описанием.

<a href="#">cipher_error</a>	Класс для обработки исключений . . . . .	7
<a href="#">modAlphaCipher</a>	Класс для шифрования и расшифрования шифра табличной маршрутной перестановки . . . . .	8





## Глава 3

# Список файлов

### 3.1 Файлы

Полный список документированных файлов.

<a href="#">modAlphaCipher.h</a>	
Описание класса <a href="#">modAlphaCipher</a>	11



## Глава 4

# Классы

### 4.1 Класс `cipher_error`

Класс для обработки исключений

```
#include <modAlphaCipher.h>
```

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

Класс для обработки исключений

Объявления и описания членов класса находятся в файле:

- [modAlphaCipher.h](#)

## 4.2 Класс modAlphaCipher

Класс для шифрования и расшифрования шифра табличной маршрутной перестановки

`#include <modAlphaCipher.h>`

## Открытые члены

- [modAlphaCipher](#) ()=delete  
Запрещающий конструктор без параметров
- [modAlphaCipher](#) (const std::wstring &skey)  
Конструктор для установки ключа
- std::wstring [encrypt](#) (const std::wstring &open\_text)  
Метод, предназначенный для шифрования шифром табличной маршрутной перестановки
- std::wstring [decrypt](#) (const std::wstring &cipher\_text)  
Метод, предназначенный для расшифрования шифра табличной маршрутной перестановки

## Закрытые члены

- std::vector< int > [convert](#) (const std::wstring &s)  
Преобразование строки в вектор
- std::wstring [convert](#) (const std::vector< int > &v)  
Преобразование вектора в строку
- std::wstring [getValidKey](#) (const std::wstring &s)
- std::wstring [getValidOpenText](#) (const std::wstring &s)
- std::wstring [getValidCipherText](#) (const std::wstring &s)

## Закрытые данные

- std::wstring [\\_convert](#)< std::codecvt\_utf8< wchar\_t >, wchar\_t > codec
- std::wstring [numAlpha](#) =L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"  
Используемый алфавит
- std::map< wchar\_t, int > [alphaNum](#)  
Ассоциативный массив "номер по символу".
- std::vector< int > [key](#)  
атрибут, хранящий ключ для шифрования или расшифрования

### 4.2.1 Подробное описание

Класс для шифрования и расшифрования шифра табличной маршрутной перестановки

### 4.2.2 Конструктор(ы)

#### 4.2.2.1 modAlphaCipher()

```
modAlphaCipher::modAlphaCipher (
    const std::wstring & skey )
```

Конструктор для установки ключа

Аргументы

целочисленное	число ключ
---------------	------------

число, которое пришло на вход записывается в "private" атрибут с названием "key"

### 4.2.3 Методы

#### 4.2.3.1 convert() [1/2]

```
std::wstring modAlphaCipher::convert (
    const std::vector< int > & v ) [inline], [private]
```

Преобразование вектора в строку

В переменную типа "wstring" с именем "result" записывается строка согласно индексам каждой буквы алфавита "numAlpha". Индексы хранятся в векторе типа "int", который поступил на вход.

```
wstring result;
for(auto i:v) {
    result.push_back(numAlpha[i]);
}
```

Возвращает

строка текста типа "wstring"

#### 4.2.3.2 convert() [2/2]

```
std::vector< int > modAlphaCipher::convert (
    const std::wstring & s )    [inline], [private]
```

##### Преобразование строки в вектор

В вектор типа "int" с именем "result" записываются числа, которые являются индексами алфавита "numAlpha", применяемый для строки, которая поступила на вход.

```
vector<int> result;
for(auto c:s) {
    result.push_back(alphaNum[c]);
}
```

##### Возвращает

std::vector <int>, в котором хранятся индексы букв сообщения из алфавита "numAlpha"

Объявления и описания членов классов находятся в файлах:

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

## Глава 5

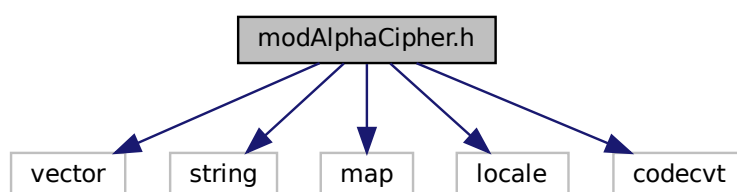
### Файлы

#### 5.1 Файл modAlphaCipher.h

Описание класса `modAlphaCipher`.

```
#include <vector>
#include <string>
#include <map>
#include <locale>
#include <codecvt>
```

Граф включаемых заголовочных файлов для modAlphaCipher.h:



#### Классы

- class `modAlphaCipher`  
Класс для шифрования и расшифрования шифра табличной маршрутной перестановки
- class `cipher_error`  
Класс для обработки исключений

### 5.1.1 Подробное описание

Описание класса [modAlphaCipher](#).

Автор

Птицин Н.М.

Версия

1.0

Дата

04.06.2021



# Предметный указатель

cipher\_error, [7](#)

convert

    modAlphaCipher, [9](#)

modAlphaCipher, [8](#)

    convert, [9](#)

    modAlphaCipher, [9](#)

modAlphaCipher.h, [11](#)