

# Шифрование методом Гронсфельда

## 1.0

Создано системой Doxygen 1.9.1

1 Иерархический список классов	1
1.1 Иерархия классов . . . . .	1
2 Алфавитный указатель классов	1
2.1 Классы . . . . .	1
3 Классы	1
3.1 Класс cipher_error . . . . .	1
3.1.1 Подробное описание . . . . .	2
3.2 Класс modAlphaCipher . . . . .	3
3.2.1 Конструктор(ы) . . . . .	3
3.2.2 Методы . . . . .	4
Предметный указатель	7

## 1 Иерархический список классов

### 1.1 Иерархия классов

Иерархия классов.

invalid_argument	
cipher_error	1
modAlphaCipher	3

## 2 Алфавитный указатель классов

### 2.1 Классы

Классы с их кратким описанием.

cipher_error	
класс исключение	1
modAlphaCipher	3

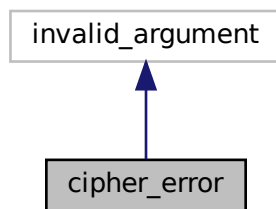
## 3 Классы

### 3.1 Класс cipher\_error

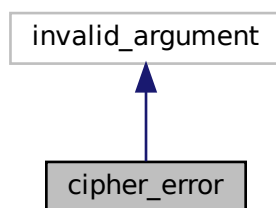
класс исключение

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

Граф наследования: cipher\_error:



Граф связей класса cipher\_error:



Открытые члены

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

### 3.1.1 Подробное описание

класс исключение

Объявления и описания членов класса находятся в файле:

- `modAlphaCipher.h`

## 3.2 Класс modAlphaCipher

### Открытые члены

- `modAlphaCipher` ()=delete  
запретим конструктор без параметров
- `modAlphaCipher` (const wstring &wskey)  
конструктор для установки ключа
- wstring `encrypt` (const wstring &open\_text)  
зашифрование
- wstring `decrypt` (const wstring &cipher\_text)  
расшифрование

### Закрытые члены

- vector< int > `convert` (const wstring &ws)  
преобразование строка-вектор
- wstring `convert` (const vector< int > &v)  
преобразование вектор-строка
- wstring `getValidKey` (const wstring &ws)  
проверка правильности ключа
- wstring `getValidOpenText` (const wstring &ws)  
проверка правильности открытого текста
- wstring `getValidCipherText` (const wstring &ws)  
проверка правильности зашифрованного текста

### Закрытые данные

- wstring\_convert< std::codecvt\_utf8< wchar\_t >, wchar\_t > codec
- wstring `numAlpha` = L"АБВГДЕЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"  
алфавит по порядку
- map< wchar\_t, int > `alphaNum`  
ассоциативный массив "номер по символу".
- vector< int > `key`  
ключ

#### 3.2.1 Конструктор(ы)

##### 3.2.1.1 modAlphaCipher() modAlphaCipher::modAlphaCipher ( const wstring & wskey )

конструктор для установки ключа

Конструктор, принимающий на вход ключ

Аргументы

ws_key	
--------	--

Возвращает

Ничего не возвращает

### 3.2.2 Методы

3.2.2.1 `decrypt()` `wstring modAlphaCipher::decrypt (`  
`const wstring & cipher_text )`

расшифрование

Метод `encrypt` дешифрованный текст.

Аргументы

cipher_text	
-------------	--

Возвращает

Дешифрованный текст

3.2.2.2 `encrypt()` `wstring modAlphaCipher::encrypt (`  
`const wstring & open_text )`

зашифрование

Метод `encrypt` зашифровывает текст.

Аргументы

open_text	
-----------	--

Возвращает

Зашифрованный текст

3.2.2.3 `getValidCipherText()` `wstring modAlphaCipher::getValidCipherText (`  
`const wstring & ws )` `[inline], [private]`

проверка правильности зашифрованного текста

Данный метод проверяет зашифрованный текст на правильность.

Аргументы

<code>ws_cipher_text</code>	
-----------------------------	--

Возвращает

Зашифрованный текст

Исключения

<code>cipher_error</code> ,если	текст пустой или невалидный
---------------------------------	-----------------------------

3.2.2.4 `getValidKey()` `std::wstring modAlphaCipher::getValidKey (`  
`const wstring & ws )` `[inline], [private]`

проверка правильности ключа

Данный метод проверяет ключ на правильность.

Аргументы

<code>ws_key</code>	
---------------------	--

Возвращает

Ключ

Исключения

<code>cipher_error</code> ,если	ключ пустой или невалидный
---------------------------------	----------------------------

3.2.2.5 `getValidOpenText()` `wstring modAlphaCipher::getValidOpenText (`  
`const wstring & ws )` `[inline], [private]`

проверка правильности открытого текста

Данный метод проверяет открытый текст на правильность. Строчные буквы превращаются в прописные. Все символы, которые не буквы удаляются.

Аргументы

<code>ws_open_text</code>	
---------------------------	--

Возвращает

Текст для расшифровки

Исключения

<code>cipher_error</code> ,если	текст пустой
---------------------------------	--------------

Объявления и описания членов классов находятся в файлах:

- `modAlphaCipher.h`
- `modAlphaCipher.cpp`

## Предметный указатель

cipher\_error, [1](#)

decrypt

modAlphaCipher, [4](#)

encrypt

modAlphaCipher, [4](#)

getValidCipherText

modAlphaCipher, [4](#)

getValidKey

modAlphaCipher, [5](#)

getValidOpenText

modAlphaCipher, [5](#)

modAlphaCipher, [3](#)

decrypt, [4](#)

encrypt, [4](#)

getValidCipherText, [4](#)

getValidKey, [5](#)

getValidOpenText, [5](#)

modAlphaCipher, [3](#)