

program1

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  |
| 4.2 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 |
| 4.2.3.3 decrypt() . . . . .            | 10 |
| 4.2.3.4 encrypt() . . . . .            | 11 |
| 4.2.3.5 getValidCipherText() . . . . . | 11 |
| 4.2.3.6 getValidKey() . . . . .        | 11 |
| 4.2.3.7 getValidOpenText() . . . . .   | 12 |
| 5 Файлы                                | 13 |
| 5.1 Файл modAlphaCipher.cpp . . . . .  | 13 |
| 5.1.1 Подробное описание . . . . .     | 14 |
| 5.2 Файл modAlphaCipher.h . . . . .    | 14 |
| 5.2.1 Подробное описание . . . . .     | 15 |
| Предметный указатель                   | 17 |



## 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.

|   |   |
|---|---|
| <code>cipher_error</code> . . . . .                                     | 7 |
| <code>modAlphaCipher</code><br>Шифрование методом Гронсфельда . . . . . | 8 |





## Глава 3

# Список файлов

### 3.1 Файлы

Полный список документированных файлов.

|  |    |
|--|----|
| <a href="#">modAlphaCipher.cpp</a>                       |    |
| файл с реализацией класса <a href="#">modAlphaCipher</a> | 13 |
| <a href="#">modAlphaCipher.h</a>                         |    |
| заголовочный файл класса <a href="#">modAlphaCipher</a>  | 14 |

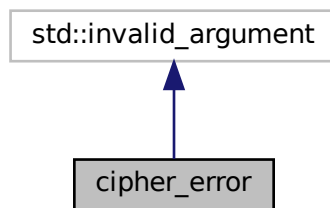


## Глава 4

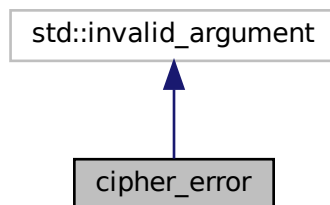
# Классы

### 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`)

Объявления и описания членов класса находятся в файле:

- [modAlphaCipher.h](#)

## 4.2 Класс modAlphaCipher

Шифрование методом Гронсфельда

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

## Открытые члены

- [modAlphaCipher](#) (`const std::wstring &skey`)  
конструктор
- `std::wstring` [encrypt](#) (`const std::wstring &open_text`)  
Зашифровывание
- `std::wstring` [decrypt](#) (`const std::wstring &cipher_text`)  
Расшифрование

## Закрытые члены

- `std::vector< int >` [convert](#) (`const std::wstring &s`)  
Преобразование
- `std::wstring` [convert](#) (`const std::vector< int > &v`)  
Преобразование
- `std::wstring` [getValidKey](#) (`const std::wstring &in`)  
Проверка ключа текста
- `std::wstring` [getValidOpenText](#) (`const std::wstring &in`)  
Проверка Откротого текста
- `std::wstring` [getValidCipherText](#) (`const std::wstring &in`)  
Проверка зашифрованного текста

## Закрытые данные

- `std::wstring` `numAlpha` = `L"АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЪЭЮЯ"`
- `std::map< wchar_t, int >` `alphaNum`
- `std::vector< int >` `key`

### 4.2.1 Подробное описание

Шифрование методом Гронсфельда

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

Предупреждения

Реализация для русского языка

### 4.2.2 Конструктор(ы)

#### 4.2.2.1 modAlphaCipher()

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

конструктор

Аргументы

|    |      |  |
|----|------|--|
| in | skey | ключ, по которому формируется зашифрованный алфавит. |
|----|------|--|

Исключения

|  |                |
|--|----------------|
| <code>cipher_error</code> ,исключение,если | ключ не верный |
|--|----------------|

### 4.2.3 Методы

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

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

Преобразование

Аргументы

|    |     |   |
|----|-----|---|
| in | str | Вектор кодов символов в локальном алфавите. |
|----|-----|---|

Возвращает

Строка по новому алфавиту.

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

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

Преобразование

Аргументы

|    |     |                 |
|----|-----|-----------------|
| in | str | Открытый текст. |
|----|-----|-----------------|

Возвращает

Вектор, состоящий из кодов символов в локальном алфавите.

#### 4.2.3.3 decrypt()

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

Расшифрование

Аргументы

|    |             |  |
|----|-------------|--|
| in | cipher_text | Зашифрованный текст. Не должен быть пустой строкой. Строчные символы автоматически преобразуются к прописным. Все символы, не являющиеся буквами удаляются |
|----|-------------|--|

Возвращает

Расшифрованная строка

Исключения

|                                     |              |
|-------------------------------------|--------------|
| <a href="#">cipher_error</a> , если | текст пустой |
|-------------------------------------|--------------|

## 4.2.3.4 encrypt()

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

Зашифровывание

Аргументы

|    |           |   |
|----|-----------|---|
| in | open_text | Открытый текст. Не должен быть пустой строкой. Строчные символы автоматически преобразуются к прописным. Все символы, не являющиеся буквами удаляются |
|----|-----------|---|

Возвращает

Зашифрованная строка

Исключения

|                                    |              |
|------------------------------------|--------------|
| <a href="#">cipher_error</a> ,если | текст пустой |
|------------------------------------|--------------|

## 4.2.3.5 getValidCipherText()

```
std::wstring modAlphaCipher::getValidCipherText (
    const std::wstring & in ) [inline], [private]
```

Проверка зашифрованного текста

Аргументы

|    |    |                     |
|----|----|---------------------|
| in | in | Зашифрованный текст |
|----|----|---------------------|

Возвращает

Зашифрованная строка

Исключения

|                                    |  |
|------------------------------------|--|
| <a href="#">cipher_error</a> ,если | текст пустой или в тексте содержатся другие символы, кроме букв алфавита |
|------------------------------------|--|

## 4.2.3.6 getValidKey()

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

Проверка ключа текста

Аргументы

|    |    |                         |
|----|----|-------------------------|
| in | in | Ключ шифрования текста. |
|----|----|-------------------------|

Возвращает

Строка с ключем

Исключения

|                                    |  |
|------------------------------------|--|
| <a href="#">cipher_error</a> ,если | ключ слишком маленький, пустой или имеет некорректные символы. |
|------------------------------------|--|

#### 4.2.3.7 getValidOpenText()

```
std::wstring modAlphaCipher::getValidOpenText (
    const std::wstring & in ) [inline], [private]
```

Проверка Откротого текста

Аргументы

|    |    |                 |
|----|----|-----------------|
| in | in | Открытый текст. |
|----|----|-----------------|

Возвращает

Строка открытого текста

Исключения

|                                    |   |
|------------------------------------|---|
| <a href="#">cipher_error</a> ,если | строка пустая или имеет некорректные символы. |
|------------------------------------|---|

Объявления и описания членов классов находятся в файлах:

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



## Глава 5

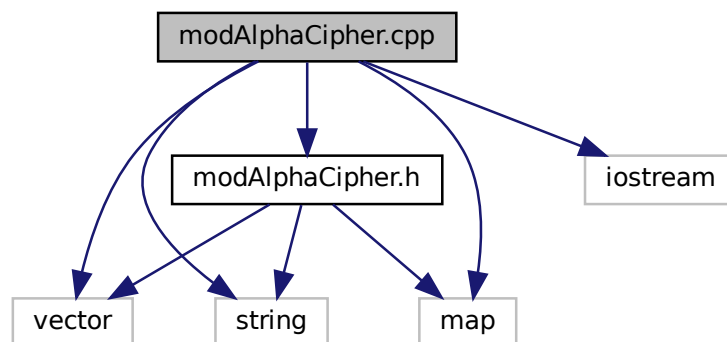
# Файлы

### 5.1 Файл modAlphaCipher.cpp

файл с реализацией класса `modAlphaCipher`.

```
#include <vector>
#include <string>
#include <map>
#include <iostream>
#include "modAlphaCipher.h"
```

Граф включаемых заголовочных файлов для `modAlphaCipher.cpp`:



### Переменные

- `std::wstring numAlpha = L"АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"`
- `std::map< wchar_t, int > alphaNum`

### 5.1.1 Подробное описание

файл с реализацией класса `modAlphaCipher`.

Автор

Жидков В.А.

Версия

1.0

Дата

29.05.2021

## 5.2 Файл `modAlphaCipher.h`

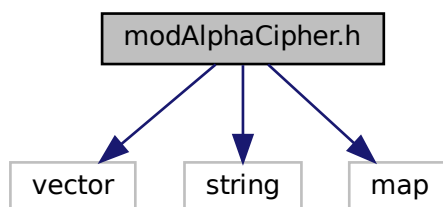
заголовочный файл класса `modAlphaCipher`.

```
#include <vector>
```

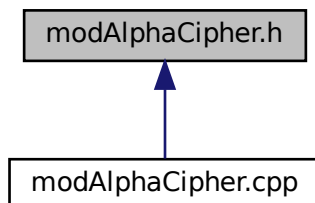
```
#include <string>
```

```
#include <map>
```

Граф включаемых заголовочных файлов для `modAlphaCipher.h`:



Граф файлов, в которые включается этот файл:



## Классы

- class `modAlphaCipher`  
Шифрование методом Гронсфельда
- class `cipher_error`

### 5.2.1 Подробное описание

заголовочный файл класса `modAlphaCipher`.

#### Автор

Жидков В.А.

#### Версия

1.0

#### Дата

29.05.2021



# Предметный указатель

- cipher\_error, [7](#)
- convert
  - modAlphaCipher, [9](#), [10](#)
- decrypt
  - modAlphaCipher, [10](#)
- encrypt
  - modAlphaCipher, [10](#)
- getValidCipherText
  - modAlphaCipher, [11](#)
- getValidKey
  - modAlphaCipher, [11](#)
- getValidOpenText
  - modAlphaCipher, [12](#)
- modAlphaCipher, [8](#)
  - convert, [9](#), [10](#)
  - decrypt, [10](#)
  - encrypt, [10](#)
  - getValidCipherText, [11](#)
  - getValidKey, [11](#)
  - getValidOpenText, [12](#)
  - modAlphaCipher, [9](#)
- modAlphaCipher.cpp, [13](#)
- modAlphaCipher.h, [14](#)