

modAlphaCipher.h

Создано системой Doxygen 1.9.4



---

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
4.1.2 Конструктор(ы) . . . . .	8
4.1.2.1 cipher_error() [1/2] . . . . .	8
4.1.2.2 cipher_error() [2/2] . . . . .	8
4.2 Класс code . . . . .	8
4.2.1 Подробное описание . . . . .	9
4.2.2 Конструктор(ы) . . . . .	9
4.2.2.1 code() . . . . .	9
4.2.3 Методы . . . . .	10
4.2.3.1 encryption() . . . . .	10
4.2.3.2 getValidCipherText() . . . . .	10
4.2.3.3 getValidKey() . . . . .	11
4.2.3.4 getValidOpenText() . . . . .	11
4.2.3.5 transcript() . . . . .	12
5 Файлы	13
5.1 Файл code.h . . . . .	13
5.1.1 Подробное описание . . . . .	13
5.2 code.h . . . . .	14
Предметный указатель	15



# Глава 1

## Иерархический список классов

### 1.1 Иерархия классов

Иерархия классов.

code . . . . .	8
invalid_argument	
cipher_error . . . . .	7



## Глава 2

# Алфавитный указатель классов

### 2.1 Классы

Классы с их кратким описанием.

<code>cipher_error</code>	Класс исключений для обработки ошибок, связанных с шифрованием . . . . .	7
<code>code</code>	Класс для шифрования и расшифрования текста с использованием числового ключа . . . . .	8





## Глава 3

# Список файлов

### 3.1 Файлы

Полный список документированных файлов.

<a href="#">code.h</a>	13
------------------------	----



## Глава 4

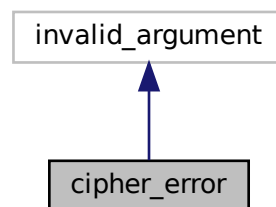
# Классы

### 4.1 Класс `cipher_error`

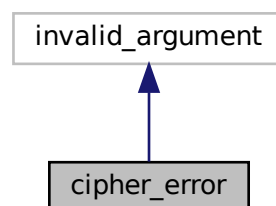
Класс исключений для обработки ошибок, связанных с шифрованием.

```
#include <code.h>
```

Граф наследования: `cipher_error`:



Граф связей класса `cipher_error`:



## Открытые члены

- [cipher\\_error](#) (const string &what\_arg)  
Конструктор, принимающий строку в качестве аргумента.
- [cipher\\_error](#) (const char \*what\_arg)  
Конструктор, принимающий C-строку в качестве аргумента.

### 4.1.1 Подробное описание

Класс исключений для обработки ошибок, связанных с шифрованием.

### 4.1.2 Конструктор(ы)

#### 4.1.2.1 cipher\_error() [1/2]

```

cipher_error::cipher_error (
    const string & what_arg ) [inline], [explicit]

```

Конструктор, принимающий строку в качестве аргумента.

Аргументы

what_arg	Описание ошибки.
----------	------------------

#### 4.1.2.2 cipher\_error() [2/2]

```

cipher_error::cipher_error (
    const char * what_arg ) [inline], [explicit]

```

Конструктор, принимающий C-строку в качестве аргумента.

Аргументы

what_arg	Описание ошибки.
----------	------------------

Объявления и описания членов класса находятся в файле:

- [code.h](#)

## 4.2 Класс code

Класс для шифрования и расшифрования текста с использованием числового ключа.

```
#include <code.h>
```

### Открытые члены

- `code()`=delete  
Конструктор по умолчанию удален.
- `code` (int skey, string text)  
Конструктор с параметрами ключа и текста.
- string `encryption` (string &text)  
Метод для шифрования текста.
- string `transcript` (string &text, string &open\_text)  
Метод для расшифрования текста.

### Закрытые члены

- int `getValidKey` (int key, const string &Text)  
Возвращает корректный ключ шифрования.
- string `getValidOpenText` (string &s)  
Проверяет и возвращает корректный открытый текст.
- string `getValidCipherText` (string &s, string &open\_text)  
Проверяет и возвращает корректный зашифрованный текст.

### Закрытые данные

- int key  
Ключ шифрования.

#### 4.2.1 Подробное описание

Класс для шифрования и расшифрования текста с использованием числового ключа.

#### 4.2.2 Конструктор(ы)

##### 4.2.2.1 code()

```
code::code (
    int skey,
    string text )
```

Конструктор с параметрами ключа и текста.

##### Аргументы

skey	Числовой ключ.
text	Текст для шифрования.

### 4.2.3 Методы

#### 4.2.3.1 encryption()

```
string code::encryption (  
    string & text )
```

Метод для шифрования текста.

Аргументы

text	Текст для шифрования.
------	-----------------------

Возвращает

Зашифрованный текст.

Исключения

<a href="#">cipher_error</a>	Если текст некорректен.
------------------------------	-------------------------

#### 4.2.3.2 getValidCipherText()

```
string code::getValidCipherText (  
    string & s,  
    string & open_text ) [inline], [private]
```

Проверяет и возвращает корректный зашифрованный текст.

Аргументы

s	Зашифрованный текст.
open_text	Открытый текст.

Возвращает

Корректный зашифрованный текст.

Исключения

<a href="#">cipher_error</a>	Если текст некорректен.
------------------------------	-------------------------

#### 4.2.3.3 getValidKey()

```
int code::getValidKey (  
    int key,  
    const string & Text )  [inline], [private]
```

Возвращает корректный ключ шифрования.

Аргументы

key	Входной ключ.
Text	Предоставленный текст.

Возвращает

Корректный ключ.

Исключения

<a href="#">cipher_error</a>	Если ключ некорректен.
------------------------------	------------------------

#### 4.2.3.4 getValidOpenText()

```
string code::getValidOpenText (  
    string & s )  [inline], [private]
```

Проверяет и возвращает корректный открытый текст.

Аргументы

s	Открытый текст.
---	-----------------

Возвращает

Корректный текст.

Исключения

<a href="#">cipher_error</a>	Если текст некорректен.
------------------------------	-------------------------

#### 4.2.3.5 transcript()

```
string code::transcript (
    string & text,
    string & open_text )
```

Метод для расшифрования текста.

Аргументы

text	Зашифрованный текст.
open_text	Открытый текст.

Возвращает

Расшифрованный текст.

Исключения

<a href="#">cipher_error</a>	Если текст некорректен.
------------------------------	-------------------------

Объявления и описания членов класса находятся в файле:

- [code.h](#)



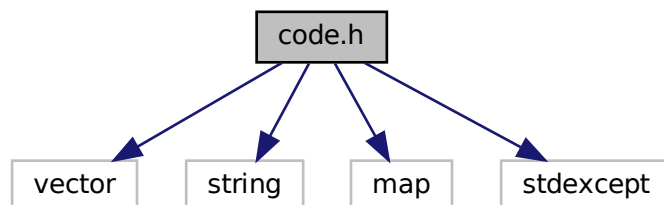
## Глава 5

# Файлы

### 5.1 Файл code.h

```
#include <vector>
#include <string>
#include <map>
#include <stdexcept>
```

Граф включаемых заголовочных файлов для code.h:



### Классы

- class `code`  
Класс для шифрования и расшифрования текста с использованием числового ключа.
- class `cipher_error`  
Класс исключений для обработки ошибок, связанных с шифрованием.

#### 5.1.1 Подробное описание

Автор

Сверчков А.Д.

Версия

2.0

Дата

25.11.2024

## 5.2 code.h

[См. документацию.](#)

```
1
6 #pragma once
7 #include <vector>
8 #include <string>
9 #include <map>
10 #include <stdexcept>
11 using namespace std;
12
16 class code {
17     private:
21         int key;
22
30         inline int getValidKey(int key, const string & Text);
31
38         inline string getValidOpenText(string & s);
39
47         inline string getValidCipherText(string & s, string & open_text);
48
49     public:
53         code() = delete;
54
60         code(int skey, string text);
61
68         string encryption(string &text);
69
77         string transcript(string &text, string &open_text);
78 };
79
83 class cipher_error: public invalid_argument {
84     public:
89         explicit cipher_error (const string& what_arg):
90             invalid_argument(what_arg) {}
91
96         explicit cipher_error (const char* what_arg):
97             invalid_argument(what_arg) {}
98 };
```

# Предметный указатель

- cipher\_error, [7](#)
  - cipher\_error, [8](#)
- code, [8](#)
  - code, [9](#)
  - encryption, [10](#)
  - getValidCipherText, [10](#)
  - getValidKey, [11](#)
  - getValidOpenText, [11](#)
  - transcript, [11](#)
- code.h, [13](#)
- encryption
  - code, [10](#)
- getValidCipherText
  - code, [10](#)
- getValidKey
  - code, [11](#)
- getValidOpenText
  - code, [11](#)
- transcript
  - code, [11](#)