

Laba_4_2

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	7
4.1.1 Подробное описание	7
4.1.2 Конструктор(ы)	7
4.1.2.1 Cipher()	8
4.1.3 Методы	9
4.1.3.1 decrypt()	9
4.1.3.2 encrypt()	9
4.1.3.3 getValidCipherText()	10
4.1.3.4 getValidKey()	10
4.1.3.5 getValidOpenText()	11
4.2 Класс cipher_error	11
4.2.1 Подробное описание	12
4.2.2 Конструктор(ы)	12
4.2.2.1 cipher_error() [1/2]	12
4.2.2.2 cipher_error() [2/2]	12
5 Файлы	13
5.1 Файл /home/stud/Загрузки/local_git/TiMP_4/Lab_4_2/modCipherBeta.h	13
5.1.1 Подробное описание	14
5.2 modCipherBeta.h	14
Предметный указатель	15

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

Cipher	7
std::invalid_argument	
cipher_error	11

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

Cipher	Класс для шифрования и расшифрования текста	7
cipher_error	Исключение, связанное с ошибками шифрования	11

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

/home/stud/Загрузки/local_git/TiMP_4/Lab_4_2/modCipherBeta.h	
Заголовочный файл для класса Cipher	13

Глава 4

Классы

4.1 Класс Cipher

Класс для шифрования и расшифрования текста.

```
#include <modCipherBeta.h>
```

Открытые члены

- `Cipher ()=delete`
Запрещает использование конструктора без параметров.
- `Cipher (const std::wstring &key_str)`
Конструктор для установки ключа.
- `wstring encrypt (const wstring &text)`
Зашифровывает текст с использованием заданного ключа.
- `wstring decrypt (const wstring &text)`
Расшифровывает зашифрованный текст с использованием заданного ключа.
- `int getValidKey (const std::wstring &key_str)`
Проверяет и возвращает валидный ключ для шифрования.
- `std::wstring getValidOpenText (const std::wstring &s)`
Проверяет и возвращает валидный открытый текст для шифрования.
- `std::wstring getValidCipherText (const std::wstring &s)`
Проверяет и возвращает валидный зашифрованный текст для расшифрования.

4.1.1 Подробное описание

Класс для шифрования и расшифрования текста.

Данный класс реализует методы шифрования и расшифрования текста на основе числового ключа. Класс обеспечивает проверку корректности ключа и текста перед выполнением операций шифрования и расшифрования.

4.1.2 Конструктор(ы)

4.1.2.1 Cipher()

```
Cipher::Cipher (  
    const std::wstring & key_str )
```

Конструктор для установки ключа.

Аргументы

key_str	Строка, представляющая ключ шифрования.
---------	---

Исключения

cipher_error	Если ключ некорректен.
------------------------------	------------------------

4.1.3 Методы

4.1.3.1 decrypt()

```
wstring Cipher::decrypt (  
    const wstring & text )
```

Расшифровывает зашифрованный текст с использованием заданного ключа.

Аргументы

text	Зашифрованный текст для расшифрования.
------	--

Возвращает

Открытый текст.

Исключения

cipher_error	Если текст некорректен.
------------------------------	-------------------------

4.1.3.2 encrypt()

```
wstring Cipher::encrypt (  
    const wstring & text )
```

Зашифровывает текст с использованием заданного ключа.

Аргументы

text	Открытый текст для шифрования.
------	--------------------------------

Возвращает

Зашифрованный текст.

Исключения

<code>cipher_error</code>	Если текст некорректен.
---------------------------	-------------------------

4.1.3.3 `getValidCipherText()`

```
std::wstring Cipher::getValidCipherText (  
    const std::wstring & s )
```

Проверяет и возвращает валидный зашифрованный текст для расшифрования.

Аргументы

<code>s</code>	Зашифрованный текст для проверки.
----------------	-----------------------------------

Возвращает

Валидный зашифрованный текст.

4.1.3.4 `getValidKey()`

```
int Cipher::getValidKey (  
    const std::wstring & key_str )
```

Проверяет и возвращает валидный ключ для шифрования.

Аргументы

<code>key_str</code>	Строка, представляющая ключ.
----------------------	------------------------------

Возвращает

Валидный ключ в виде целого числа.

Исключения

<code>cipher_error</code>	Если ключ некорректен.
---------------------------	------------------------

4.1.3.5 getValidOpenText()

```
std::wstring Cipher::getValidOpenText (
    const std::wstring & s )
```

Проверяет и возвращает валидный открытый текст для шифрования.

Аргументы

s	Открытый текст для проверки.
---	------------------------------

Возвращает

Валидный открытый текст.

Объявления и описания членов класса находятся в файле:

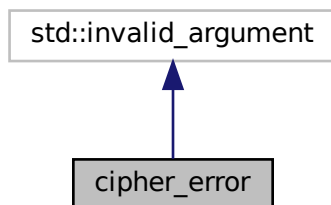
- /home/stud/Загрузки/local_git/TiMP_4/Lab_4_2/[modCipherBeta.h](#)

4.2 Класс cipher_error

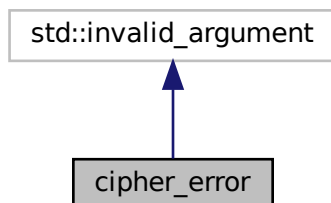
Исключение, связанное с ошибками шифрования.

```
#include <modCipherBeta.h>
```

Граф наследования: cipher_error:



Граф связей класса cipher_error:



Открытые члены

- `cipher_error` (`const std::string &what_arg`)
Конструктор исключения с сообщением об ошибке.
- `cipher_error` (`const char *what_arg`)
Конструктор исключения с сообщением об ошибке.

4.2.1 Подробное описание

Исключение, связанное с ошибками шифрования.

Данный класс представляет собой пользовательское исключение, которое наследуется от `std::invalid_argument` и используется для обработки ошибок, возникающих при работе с классом `Cipher`.

4.2.2 Конструктор(ы)

4.2.2.1 `cipher_error()` [1/2]

```
cipher_error::cipher_error (
    const std::string & what_arg ) [explicit]
```

Конструктор исключения с сообщением об ошибке.

Аргументы

<code>what_arg</code>	Сообщение об ошибке в виде строки.
-----------------------	------------------------------------

4.2.2.2 `cipher_error()` [2/2]

```
cipher_error::cipher_error (
    const char * what_arg ) [explicit]
```

Конструктор исключения с сообщением об ошибке.

Аргументы

<code>what_arg</code>	Сообщение об ошибке в виде C-строки.
-----------------------	--------------------------------------

Объявления и описания членов класса находятся в файле:

- `/home/stud/Загрузки/local_git/TiMP_4/Lab_4_2/modCipherBeta.h`

Глава 5

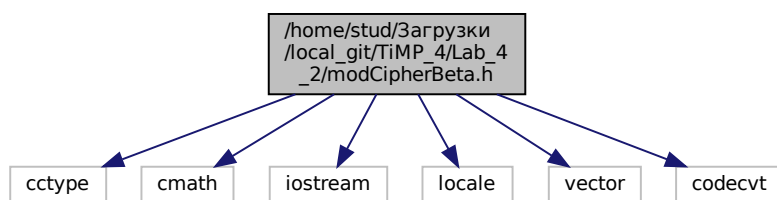
Файлы

5.1 Файл /home/stud/Загрузки/local_git/TiMP_4/Lab_4_2/modCipherBeta.h

Заголовочный файл для класса [Cipher](#).

```
#include <cctype>
#include <cmath>
#include <iostream>
#include <locale>
#include <vector>
#include <codecvt>
```

Граф включаемых заголовочных файлов для modCipherBeta.h:



Классы

- class [Cipher](#)
Класс для шифрования и расшифрования текста.
- class [cipher_error](#)
Исключение, связанное с ошибками шифрования.

5.1.1 Подробное описание

Заголовочный файл для класса [Cipher](#).

Автор

Мочалов Ю.А.

Версия

1.0

Дата

09.12.2024

Данный файл содержит определение класса [Cipher](#), который реализует методы шифрования и расшифрования текста с использованием заданного ключа.

5.2 modCipherBeta.h

[См. документацию.](#)

```
1 #pragma once
2
3 #include <cctype>
4 #include <cmath>
5 #include <iostream>
6 #include <locale>
7 #include <vector>
8 #include <codecvt>
9
29 class Cipher
30 {
31 private:
32     int key;
33
34 public:
38     Cipher() = delete;
39
45     Cipher(const std::wstring& key_str);
46
53     wstring encrypt(const wstring& text);
54
61     wstring decrypt(const wstring& text);
62
69     int getValidKey(const std::wstring& key_str);
70
76     std::wstring getValidOpenText(const std::wstring& s);
77
83     std::wstring getValidCipherText(const std::wstring& s);
84 };
85
94 class cipher_error : public std::invalid_argument
95 {
96 public:
101     explicit cipher_error(const std::string& what_arg);
102
107     explicit cipher_error(const char* what_arg);
108 };
```

Предметный указатель

[/home/stud/Загрузки/local_git/TiMP_4/Lab_4_2/modCipherBeta.h](#),
[13](#), [14](#)

Cipher, [7](#)

 Cipher, [7](#)

 decrypt, [9](#)

 encrypt, [9](#)

 getValidCipherText, [10](#)

 getValidKey, [10](#)

 getValidOpenText, [10](#)

cipher_error, [11](#)

 cipher_error, [12](#)

decrypt

 Cipher, [9](#)

encrypt

 Cipher, [9](#)

getValidCipherText

 Cipher, [10](#)

getValidKey

 Cipher, [10](#)

getValidOpenText

 Cipher, [10](#)