

Laba_4_1

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 modAlphaCipher	9
4.2.1 Подробное описание	9
4.2.2 Конструктор(ы)	9
4.2.2.1 modAlphaCipher()	9
4.2.3 Методы	10
4.2.3.1 decrypt()	10
4.2.3.2 encrypt()	10
5 Файлы	11
5.1 Файл /home/stud/Загрузки/local_git/TiMP_4/Lab_4_1/modAlphaCipher.h	11
5.1.1 Подробное описание	12
5.2 modAlphaCipher.h	12
Предметный указатель	13

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

std::invalid_argument	
cipher_error	7
modAlphaCipher	9

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	Исключение, связанное с ошибками шифрования	7
modAlphaCipher	Класс для шифрования и расшифрования текста с использованием модифицированного алфавитного шифра	9

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

/home/stud/Загрузки/local_git/TiMP_4/Lab_4_1/modAlphaCipher.h	
Заголовочный файл для класса modAlphaCipher	11

Глава 4

Классы

4.1 Класс `cipher_error`

Исключение, связанное с ошибками шифрования.

```
#include <modAlphaCipher.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.

4.1.1 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 `modAlphaCipher`.

4.1.2 Конструктор(ы)

4.1.2.1 `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.
-----------------------	------------------------------------

4.1.2.2 `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_1/modAlphaCipher.h`

4.2 Класс modAlphaCipher

Класс для шифрования и расшифрования текста с использованием модифицированного алфавитного шифра.

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

Открытые члены

- `modAlphaCipher ()=delete`
Запрещает использование конструктора без параметров.
- `modAlphaCipher (const std::wstring &skey)`
Конструктор для установки ключа.
- `std::wstring encrypt (const std::wstring &open_text)`
Зашифровывает открытый текст с использованием заданного ключа.
- `std::wstring decrypt (const std::wstring &cipher_text)`
Расшифровывает зашифрованный текст с использованием заданного ключа.

4.2.1 Подробное описание

Класс для шифрования и расшифрования текста с использованием модифицированного алфавитного шифра.

Данный класс реализует шифрование и расшифрование текста на основе заданного ключа, используя ассоциативный массив для сопоставления символов алфавита с их номерами.

4.2.2 Конструктор(ы)

4.2.2.1 modAlphaCipher()

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

Конструктор для установки ключа.

Аргументы

<code>skey</code>	Строка, представляющая ключ шифрования.
-------------------	-----------------------------------------

Исключения

<code>cipher_error</code>	текст содержит недопустимые символы.
---------------------------	--------------------------------------

4.2.3 Методы

4.2.3.1 decrypt()

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

Расшифровывает зашифрованный текст с использованием заданного ключа.

Аргументы

<code>cipher_text</code>	Зашифрованный текст для расшифрования.
--------------------------	----------------------------------------

Возвращает

Открытый текст.

Исключения

<code>cipher_error</code>	текст содержит недопустимые символы.
---------------------------	--------------------------------------

4.2.3.2 encrypt()

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

Зашифровывает открытый текст с использованием заданного ключа.

Аргументы

<code>open_text</code>	Открытый текст для шифрования.
------------------------	--------------------------------

Возвращает

Зашифрованный текст.

Исключения

<code>cipher_error</code>	текст содержит недопустимые символы.
---------------------------	--------------------------------------

Объявления и описания членов класса находятся в файле:

- `/home/stud/Загрузки/local_git/TiMP_4/Lab_4_1/modAlphaCipher.h`

Глава 5

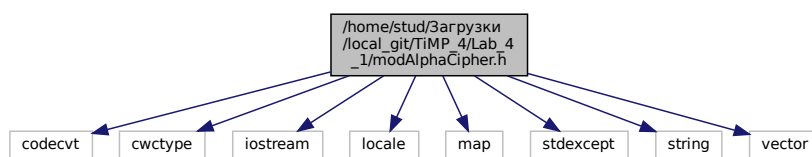
Файлы

5.1 Файл /home/stud/Загрузки/local_git/TiMP_4/Lab_4_1/modAlphaCipher.h

Заголовочный файл для класса `modAlphaCipher`.

```
#include <codecvt>
#include <cwctype>
#include <iostream>
#include <locale>
#include <map>
#include <stdexcept>
#include <string>
#include <vector>
```

Граф включаемых заголовочных файлов для `modAlphaCipher.h`:



Классы

- class `modAlphaCipher`
Класс для шифрования и расшифрования текста с использованием модифицированного алфавитного шифра.
- class `cipher_error`
Исключение, связанное с ошибками шифрования.

5.1.1 Подробное описание

Заголовочный файл для класса `modAlphaCipher`.

Автор

Мочалов Ю.А.

Версия

1.0

Дата

09.12.2024

Данный файл содержит определение класса `modAlphaCipher`, который реализует методы шифрования и расшифрования текста с использованием заданного ключа.

5.2 `modAlphaCipher.h`

См. документацию.

```
1 #pragma once
2
3 #include <codecvt>
4 #include <cwctype>
5 #include <iostream>
6 #include <locale>
7 #include <map>
8 #include <stdexcept>
9 #include <string>
10 #include <vector>
11
12 class modAlphaCipher
13 {
14 private:
15     std::wstring numAlpha; // Алфавит по порядку
16     std::map<char, int> alphaNum; // Ассоциативный массив "номер по символу"
17     std::vector<int> key; // Ключ для шифрования/расшифрования
18
19     std::vector<int> convert(const std::wstring& s);
20     std::wstring convert(const std::vector<int>& v);
21     std::wstring getValidKey(const std::wstring& s);
22     std::wstring getValidOpenText(const std::wstring& s);
23     std::wstring getValidCipherText(const std::wstring& s);
24 public:
25     modAlphaCipher() = delete;
26     modAlphaCipher(const std::wstring& skey);
27
28     std::wstring encrypt(const std::wstring& open_text);
29     std::wstring decrypt(const std::wstring& cipher_text);
30 };
31
32 class cipher_error : public std::invalid_argument
33 {
34 public:
35     explicit cipher_error(const std::string& what_arg);
36     explicit cipher_error(const char* what_arg);
37 };
38
```


Предметный указатель

`/home/stud/Загрузки/local_git/TiMP_4/Lab_4_1/modAlphaCipher.h,`
[11](#), [12](#)

`cipher_error,` [7](#)
 `cipher_error,` [8](#)

`decrypt`
 `modAlphaCipher,` [10](#)

`encrypt`
 `modAlphaCipher,` [10](#)

`modAlphaCipher,` [9](#)
 `decrypt,` [10](#)
 `encrypt,` [10](#)
 `modAlphaCipher,` [9](#)