

Лаб. работа №4. Шифр Гронсвельда.

Создано системой Doxygen 1.9.4

1 Алфавитный указатель классов	1
1.1 Классы	1
2 Список файлов	3
2.1 Файлы	3
3 Классы	5
3.1 Класс cipher_error	5
3.1.1 Подробное описание	5
3.1.2 Конструктор(ы)	5
3.1.2.1 cipher_error() [1/2]	5
3.1.2.2 cipher_error() [2/2]	6
3.1.3 Методы	6
3.1.3.1 what()	6
3.1.4 Данные класса	6
3.1.4.1 msg	6
3.2 Класс modAlphaCipher	7
3.2.1 Подробное описание	7
3.2.2 Конструктор(ы)	7
3.2.2.1 modAlphaCipher() [1/2]	8
3.2.2.2 modAlphaCipher() [2/2]	8
3.2.3 Методы	9
3.2.3.1 convert() [1/2]	9
3.2.3.2 convert() [2/2]	9
3.2.3.3 decrypt()	9
3.2.3.4 encrypt()	10
3.2.3.5 getValidCipherText()	10
3.2.3.6 getValidKey()	10
3.2.3.7 getValidOpenText()	11
3.2.4 Данные класса	11
3.2.4.1 alphaNum	11
3.2.4.2 key	11
3.2.4.3 numAlpha	11
4 Файлы	13
4.1 Файл main.cpp	13
4.1.1 Функции	13
4.1.1.1 isValid()	14
4.1.1.2 main()	14
4.2 Файл modAlphaCipher.cpp	14
4.3 Файл modAlphaCipher.h	14
4.3.1 Подробное описание	15
4.4 modAlphaCipher.h	16
4.5 Файл modException.h	16

4.5.1 Подробное описание	17
4.6 modException.h	17
Предметный указатель	19

Глава 1

Алфавитный указатель классов

1.1 Классы

Классы с их кратким описанием.

cipher_error	Класс для ошибок	5
modAlphaCipher	Класс для зашифрования и расшифрования текста	7

Глава 2

Список файлов

2.1 Файлы

Полный список файлов.

main.cpp	13
modAlphaCipher.cpp	14
modAlphaCipher.h	
Файл с описание класса modAlphaCipher	14
modException.h	
Файл с описанием класса ошибок	16

Глава 3

Классы

3.1 Класс cipher_error

Класс для ошибок

```
#include <modException.h>
```

Открытые члены

- `cipher_error` (const std::wstring &what_arg)
Конструктор со строкой.
- `cipher_error` (const wchar_t *what_arg)
Конструктор с массивом символов.
- std::wstring `what` () const
Получить сообщение.

Закрытые данные

- std::wstring `msg`
Сообщение об ошибке.

3.1.1 Подробное описание

Класс для ошибок

3.1.2 Конструктор(ы)

3.1.2.1 cipher_error() [1/2]

```
cipher_error::cipher_error (  
    const std::wstring & what_arg ) [inline]
```

Конструктор со строкой.

Аргументы

in	what_arg	Сообщение.
----	----------	------------

3.1.2.2 cipher_error() [2/2]

```

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

```

Конструктор с массивом символов.

Аргументы

in	what_arg	Сообщение.
----	----------	------------

3.1.3 Методы

3.1.3.1 what()

```

std::wstring cipher_error::what ( ) const [inline]

```

Получить сообщение.

Возвращает

std::wstring Сообщение.

3.1.4 Данные класса

3.1.4.1 msg

```

std::wstring cipher_error::msg [private]

```

Сообщение об ошибке.

Объявления и описания членов класса находятся в файле:

- [modException.h](#)

3.2 Класс modAlphaCipher

Класс для зашифрования и расшифрования текста.

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

Открытые члены

- `modAlphaCipher` ()=delete
- `modAlphaCipher` (const std::wstring &skey)
Конструктор с ключом.
- std::wstring `encrypt` (const std::wstring &open_text)
Зашифровать текст.
- std::wstring `decrypt` (const std::wstring &cipher_text)
Расшифровать текст.

Закрытые члены

- std::vector< int > `convert` (const std::wstring &s)
Преобразовать строку в вектор.
- std::wstring `convert` (const std::vector< int > &v)
Преобразовать вектор в строку.
- std::wstring `getValidKey` (const std::wstring &s)
Проверить и получить ключ.
- std::wstring `getValidOpenText` (const std::wstring &s)
Проверить и получить открытый текст.
- std::wstring `getValidCipherText` (const std::wstring &s)
Проверить и получить шифротекст.

Закрытые данные

- std::wstring `numAlpha` = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"
алфавит по порядку
- std::map< char, int > `alphaNum`
ассоциативный массив "номер по символу"
- std::vector< int > `key`

3.2.1 Подробное описание

Класс для зашифрования и расшифрования текста.

3.2.2 Конструктор(ы)

3.2.2.1 modAlphaCipher() [1/2]

modAlphaCipher::modAlphaCipher () [delete]

3.2.2.2 modAlphaCipher() [2/2]

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

Конструктор с ключом.

Аргументы

in	skey	Ключ.
----	------	-------

3.2.3 Методы

3.2.3.1 convert() [1/2]

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

Преобразовать вектор в строку.

Аргументы

in	v	Вектор индексов.
----	---	------------------

Возвращает

std::vector<int> Строка.

3.2.3.2 convert() [2/2]

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

Преобразовать строку в вектор.

Аргументы

in	s	Строка.
----	---	---------

Возвращает

std::vector<int> Вектор индексов.

3.2.3.3 decrypt()

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

Расшифровать текст.

Аргументы

in	cipher_text	Зашифрованный текст
----	-------------	---------------------

Возвращает

std::wstring Открытый текст.

3.2.3.4 encrypt()

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

Зашифровать текст.

Аргументы

in	open_text	Открытый текст.
----	-----------	-----------------

Возвращает

std::wstring Зашифрованный текст.

3.2.3.5 getValidCipherText()

```
std::wstring modAlphaCipher::getValidCipherText (
    const std::wstring & s ) [private]
```

Проверить и получить шифротекст.

Аргументы

in	s	Шифротекст.
----	---	-------------

Возвращает

std::wstring Обработанная строка.

3.2.3.6 getValidKey()

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

Проверить и получить ключ.

Аргументы

in	s	Строка-ключ.
----	---	--------------

Возвращает

std::wstring Обработанный ключ.

3.2.3.7 getValidOpenText()

```
std::wstring modAlphaCipher::getValidOpenText (  
    const std::wstring & s ) [private]
```

Проверить и получить открытый текст.

Аргументы

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

Возвращает

std::wstring Обработанная строка.

3.2.4 Данные класса

3.2.4.1 alphaNum

```
std::map<char,int> modAlphaCipher::alphaNum [private]
```

ассоциативный массив "номер по символу"

3.2.4.2 key

```
std::vector<int> modAlphaCipher::key [private]
```

3.2.4.3 numAlpha

```
std::wstring modAlphaCipher::numAlpha = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ" [private]
```

алфавит по порядку

Объявления и описания членов классов находятся в файлах:

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

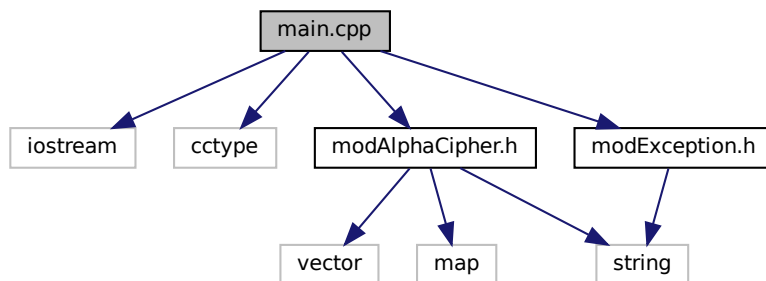
Глава 4

Файлы

4.1 Файл main.cpp

```
#include <iostream>
#include <cctype>
#include "modAlphaCipher.h"
#include "modException.h"
```

Граф включаемых заголовочных файлов для main.cpp:



Функции

- bool `isValid` (const wstring &s)
- int `main` (int argc, char **argv)

4.1.1 Функции

4.1.1.1 isValid()

```
bool isValid (  
    const wstring & s )
```

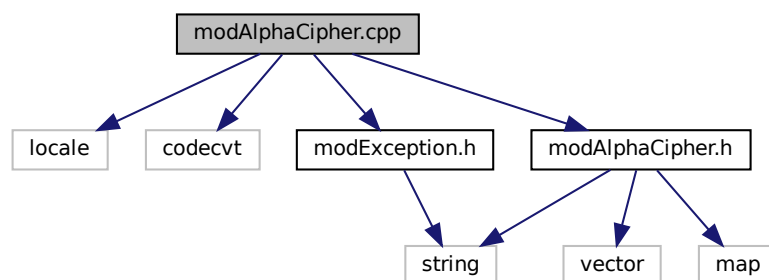
4.1.1.2 main()

```
int main (  
    int argc,  
    char ** argv )
```

4.2 Файл modAlphaCipher.cpp

```
#include <locale>  
#include <codecvt>  
#include "modException.h"  
#include "modAlphaCipher.h"
```

Граф включаемых заголовочных файлов для modAlphaCipher.cpp:



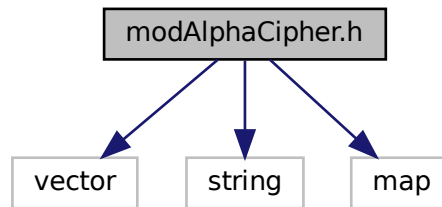
4.3 Файл modAlphaCipher.h

Файл с описанием класса [modAlphaCipher](#).

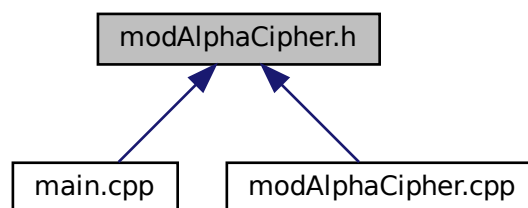
```
#include <vector>  
#include <string>
```

```
#include <map>
```

Граф включаемых заголовочных файлов для modAlphaCipher.h:



Граф файлов, в которые включается этот файл:



Классы

- class `modAlphaCipher`
Класс для зашифрования и расшифрования текста.

4.3.1 Подробное описание

Файл с описанием класса `modAlphaCipher`.

Автор

Шурыгин Д.Д.

Дата

2023-12-25

4.4 modAlphaCipher.h

См. документацию.

```

1
7 #pragma once
8
9 #include <vector>
10 #include <string>
11 #include <map>
12
16 class modAlphaCipher {
17     private:
18         std::wstring numAlpha = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ";
19         std::map <char,int> alphaNum;
20         std::vector <int> key; // ключ
27         std::vector<int> convert(const std::wstring& s);
34         std::wstring convert(const std::vector<int>& v);
41         std::wstring getValidKey(const std::wstring & s);
48         std::wstring getValidOpenText(const std::wstring & s);
55         std::wstring getValidCipherText(const std::wstring & s);
56     public:
57         modAlphaCipher()=delete;
63         modAlphaCipher(const std::wstring& skey);
70         std::wstring encrypt(const std::wstring& open_text);
77         std::wstring decrypt(const std::wstring& cipher_text);
78 };

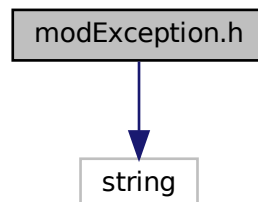
```

4.5 Файл modException.h

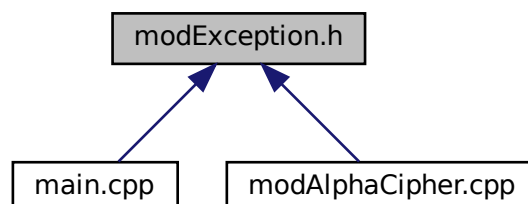
Файл с описанием класса ошибок

```
#include <string>
```

Граф включаемых заголовочных файлов для modException.h:



Граф файлов, в которые включается этот файл:



Классы

- class `cipher_error`
Класс для ошибок

4.5.1 Подробное описание

Файл с описанием класса ошибок

Автор

Шурыгин Д.Д.

Дата

2023-12-25

4.6 modException.h

[См. документацию.](#)

```
1
7 #include <string>
8
12 class cipher_error {
13 private:
14     std::wstring msg;
15 public:
21     cipher_error(const std::wstring& what_arg): msg(what_arg) { }
27     explicit cipher_error(const wchar_t* what_arg): msg(what_arg) { }
33     std::wstring what()const { return msg; }
34 };
```


Предметный указатель

alphaNum
 modAlphaCipher, [11](#)

cipher_error, [5](#)
 cipher_error, [5](#), [6](#)
 msg, [6](#)
 what, [6](#)

convert
 modAlphaCipher, [9](#)

decrypt
 modAlphaCipher, [9](#)

encrypt
 modAlphaCipher, [10](#)

getValidCipherText
 modAlphaCipher, [10](#)

getValidKey
 modAlphaCipher, [10](#)

getValidOpenText
 modAlphaCipher, [11](#)

isValid
 main.cpp, [13](#)

key
 modAlphaCipher, [11](#)

main
 main.cpp, [14](#)

main.cpp, [13](#)
 isValid, [13](#)
 main, [14](#)

modAlphaCipher, [7](#)
 alphaNum, [11](#)
 convert, [9](#)
 decrypt, [9](#)
 encrypt, [10](#)
 getValidCipherText, [10](#)
 getValidKey, [10](#)
 getValidOpenText, [11](#)
 key, [11](#)
 modAlphaCipher, [7](#), [8](#)
 numAlpha, [11](#)

modAlphaCipher.cpp, [14](#)

modAlphaCipher.h, [14](#)

modException.h, [16](#)

msg
 cipher_error, [6](#)

numAlpha
 modAlphaCipher, [11](#)

what
 cipher_error, [6](#)