

modAlphaCipher Демина
1.0

Создано системой Doxygen 1.9.1

1 Иерархический список классов	1
1 Иерархический список классов	1
1.1 Иерархия классов	1
2 Алфавитный указатель классов	1
2.1 Классы	1
3 Список файлов	2
3.1 Файлы	2
4 Классы	2
4.1 Класс cipher_error	2
4.1.1 Подробное описание	3
4.2 Класс modAlphaCipher	3
4.2.1 Подробное описание	4
4.2.2 Методы	4
5 Файлы	7
5.1 Файл main.cpp	7
5.1.1 Подробное описание	7
5.2 Файл modAlphaCipher.cpp	8
5.2.1 Подробное описание	8
5.3 Файл modAlphaCipher.h	9
5.3.1 Подробное описание	9
Предметный указатель	11

1 Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

std::invalid_argument	
cipher_error	2
modAlphaCipher	3

2 Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	
Класс ошибок для шифрования методом Гронсфельда	2

[modAlphaCipher](#)

Шифрование методом Гронсфельда

3

3 Список файлов

3.1 Файлы

Полный список документированных файлов.

[main.cpp](#)

Главный файл для модуля Gronsfeld

7

[modAlphaCipher.cpp](#)

Исполняемый файл для модуля Gronsfeld

8

[modAlphaCipher.h](#)

Заголовочный файл для модуля Gronsfeld

9

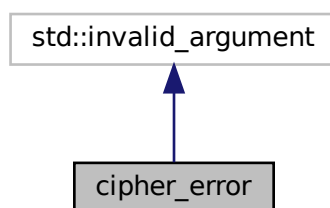
4 Классы

4.1 Класс cipher_error

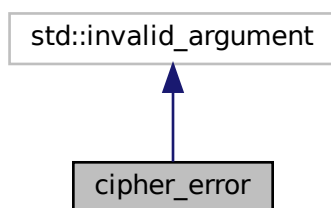
Класс ошибок для шифрования методом Гронсфельда

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

Граф наследования: cipher_error:



Граф связей класса cipher_error:



Открытые члены

- cipher_error (const std::string &what_arg)
- cipher_error (const char *what_arg)

4.1.1 Подробное описание

Класс ошибок для шифрования методом Гронсфельда

Используется для обозначения специфических ошибок, возникающих при работе с модулем

Предупреждения

Вывод сообщений об ошибках только на английском языке

Объявления и описания членов класса находятся в файле:

- [modAlphaCipher.h](#)

4.2 Класс modAlphaCipher

Шифрование методом Гронсфельда

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

Открытые члены

- [modAlphaCipher](#) ()=delete
Запретим конструктор без параметров
- [modAlphaCipher](#) (const std::wstring &skey)
Конструктор для установки ключа
- std::wstring [encrypt](#) (const std::wstring &open_text)
Зашифрование
- std::wstring [decrypt](#) (const std::wstring &cipher_text)
Расшифрование

Закрытые члены

- `std::vector< int > convert (const std::wstring &s)`
Преобразование строка-вектор
- `std::wstring convert (const std::vector< int > &v)`
Преобразование вектор-строка
- `std::wstring getValidKey (const std::wstring &s)`
Получение валидного ключа
- `std::wstring getValidOpenText (const std::wstring &s)`
Получение валидного открытого текста
- `std::wstring getValidCipherText (const std::wstring &s)`
Получение валидного шифротекста

Закрытые данные

- `std::wstring numAlpha = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"`
Русский алфавит по порядку
- `std::map< wchar_t, int > alphaNum`
Ассоциативный массив "номер по символу".
- `std::vector< int > key`
Ключ

4.2.1 Подробное описание

Шифрование методом Гронсфельда

Ключ устанавливается в конструкторе. Расшифрование и зашифрование просходит с помощью методов `encrypt` и `decrypt`

Предупреждения

Реализована поддержка только для русского языка

4.2.2 Методы

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

Расшифрование

Строчные символы шифротекста автоматически преобразуются к заглавным. Все не-буквы удаляются

Аргументы

in	cipher_text	Шифро текст. Строка не должна быть пустой.
----	-------------	--

Возвращает

Расшифрованная строка

Исключения

<code>cipher_error</code> ,если	текст пустой или присутствуют символы, не входящие в заданный алфавит
---------------------------------	---

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

Зашифрование

Строчные символы открытого текста автоматически преобразуются к заглавным. Все не-буквы удаляются

Аргументы

<code>in</code>	<code>open_text</code>	Открытый текст. Строка не должна быть пустой.
-----------------	------------------------	---

Возвращает

Зашифрованная строка

Исключения

<code>cipher_error</code> ,если	открытый текст пустой или присутствуют символы, не входящие в заданный алфавит
---------------------------------	--

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

Получение валидного шифротекста

Строчные символы шифротекста автоматически преобразуются к заглавным. Все не-буквы удаляются

Аргументы

<code>in</code>	<code>cipher_text</code>	Шифро текст. Строка не должна быть пусой
-----------------	--------------------------	--

Возвращает

Обработанный шифро текст

Исключения

cipher_error ,если	шифротекст пустой или присутствуют символы, не входящие в заданный алфавит
------------------------------------	--

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

Получение валидного ключа

Строчные символы ключа автоматически преобразуются к заглавным. Все не-буквы удаляются

Аргументы

<code>in</code>	<code>key</code>	Ключ. Строка не должна быть пустой
-----------------	------------------	------------------------------------

Возвращает

Обработанный ключ

Исключения

cipher_error ,если	ключ пустой или присутствуют символы, не входящие в заданный алфавит
------------------------------------	--

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

Получение валидного открытого текста

Строчные символы текста автоматически преобразуются к заглавным. Все не-буквы удаляются

Аргументы

<code>in</code>	<code>open_text</code>	Открытый текст. Строка не должна быть пустой
-----------------	------------------------	--

Возвращает

Обработанный открытый текст

Исключения

cipher_error ,если	открытый текст пустой или присутствуют символы, не входящие в заданный алфавит
------------------------------------	--

Объявления и описания членов классов находятся в файлах:

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

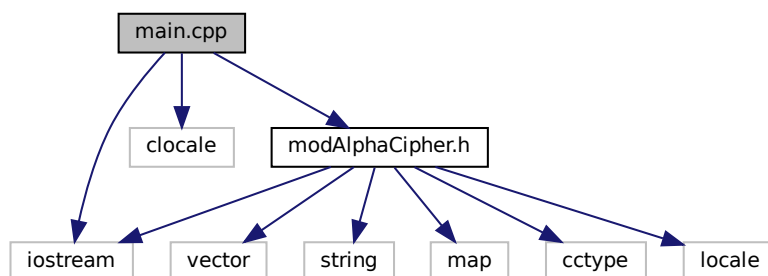
5 Файлы

5.1 Файл main.cpp

Главный файл для модуля Gronsfeld.

```
#include <iostream>
#include <clocale>
#include "modAlphaCipher.h"
```

Граф включаемых заголовочных файлов для main.cpp:



Функции

- void key_validation (wstring &key, const wstring &text)
- int main (int argc, char **argv)

5.1.1 Подробное описание

Главный файл для модуля Gronsfeld.

Автор

Демина К.С

Версия

1.0

Дата

18.01.24

Авторство

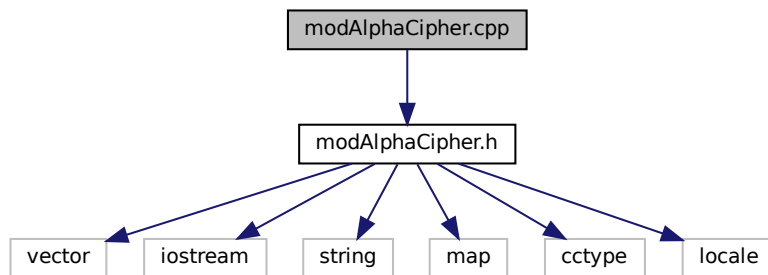
ИБСТ ПГУ

5.2 Файл modAlphaCipher.cpp

Исполняемый файл для модуля Gronsfeld.

```
#include "modAlphaCipher.h"
```

Граф включаемых заголовочных файлов для modAlphaCipher.cpp:



5.2.1 Подробное описание

Исполняемый файл для модуля Gronsfeld.

Автор

Демина К.С

Версия

1.0

Дата

18.01.24

Авторство

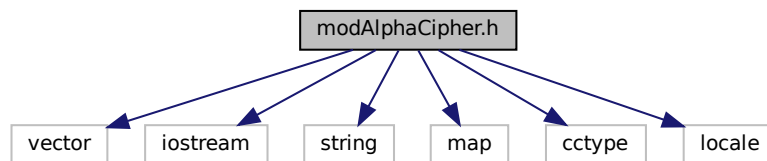
ИБСТ ПГУ

5.3 Файл modAlphaCipher.h

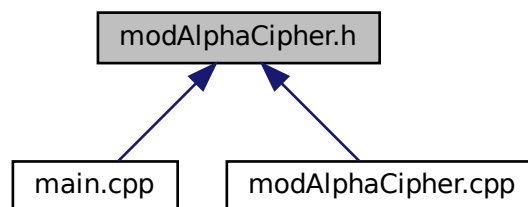
Заголовочный файл для модуля Gronsfeld.

```
#include <vector>
#include <iostream>
#include <string>
#include <map>
#include <cctype>
#include <locale>
```

Граф включаемых заголовочных файлов для modAlphaCipher.h:



Граф файлов, в которые включается этот файл:



Классы

- class `modAlphaCipher`
Шифрование методом Гронсфельда
- class `cipher_error`
Класс ошибок для шифрования методом Гронсфельда

5.3.1 Подробное описание

Заголовочный файл для модуля Gronsfeld.

Автор

Демина К.С.

Версия

1.0

Дата

18.01.24

Авторство

ИБСТ ПГУ

Предметный указатель

cipher_error, [2](#)

decrypt

modAlphaCipher, [4](#)

encrypt

modAlphaCipher, [5](#)

getValidCipherText

modAlphaCipher, [5](#)

getValidKey

modAlphaCipher, [6](#)

getValidOpenText

modAlphaCipher, [6](#)

main.cpp, [7](#)

modAlphaCipher, [3](#)

decrypt, [4](#)

encrypt, [5](#)

getValidCipherText, [5](#)

getValidKey, [6](#)

getValidOpenText, [6](#)

modAlphaCipher.cpp, [8](#)

modAlphaCipher.h, [9](#)