

Kalgudaeva

1.0

Создано системой Doxygen 1.9.1

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
4.2 Класс modAlphaCipher	8
4.2.1 Подробное описание	9
4.2.2 Методы	9
4.2.2.1 decrypt()	9
4.2.2.2 encrypt()	9
4.2.2.3 getValidCipherText()	10
4.2.2.4 getValidKey()	10
4.2.2.5 getValidOpenText()	10
5 Файлы	13
5.1 Файл modAlphaCipher.h	13
5.1.1 Подробное описание	13
Предметный указатель	15

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

std::invalid_argument	
cipher_error	7
modAlphaCipher	8

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	7
modAlphaCipher Шифрование методом Гронсфельда	8

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

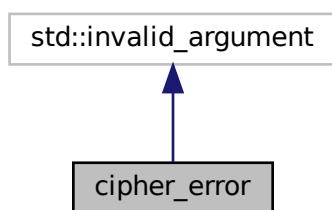
modAlphaCipher.h	
Класс-исключение	13

Глава 4

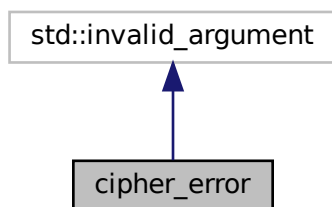
Классы

4.1 Класс cipher_error

Граф наследования: cipher_error:



Граф связей класса cipher_error:



Открытые члены

- `cipher_error` (const std::string &what_arg)
Принимает строку, поднимает исключение
- `cipher_error` (const char *what_arg)
Принимает строку, поднимает исключение

Объявления и описания членов класса находятся в файле:

- `modAlphaCipher.h`

4.2 Класс modAlphaCipher

Шифрование методом Гронсфелда

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

Открытые члены

- `modAlphaCipher` ()=delete
запретим конструктор без параметров
- `modAlphaCipher` (const std::wstring &wskey)
конструктор для установки ключа
- std::wstring `encrypt` (const std::wstring &open_text)
Зашифровывание Открытый текст не должен быть пустой строкой. Строчные символы автоматически преобразуются к прописным. Все не-буквы удаляются
- std::wstring `decrypt` (const std::wstring &cipher_text)
Расшифровывание

Закрытые члены

- std::vector< int > `convert` (const std::wstring &ws)
Преобразование строка-вектор
- std::wstring `convert` (const std::vector< int > &v)
Преобразование вектор-строка
- std::wstring `getValidKey` (const std::wstring &ws)
Проверка на хороший ключ
- std::wstring `getValidOpenText` (const std::wstring &ws)
Метод для валидации открытого текста
Все символы не принадлежащие алфавиту игнорируются
Все строчные символы преобразуются к прописным
- std::wstring `getValidCipherText` (const std::wstring &ws)
Метод для валидации текста зашифрования

Закрытые данные

- `std::wstring_convert< std::codecvt_utf8< wchar_t >, wchar_t > codec`
- `std::wstring numAlpha = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"`
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. Для зашифровывания и расшифровывания предназначены методы `encrypt` и `decrypt`.

Предупреждения

Реализация только для русского языка

4.2.2 Методы

4.2.2.1 `decrypt()`

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

Расшифровывание

Исключения

<code>cipher_error</code> , если	текст пустой
----------------------------------	--------------

4.2.2.2 `encrypt()`

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

Зашифровывание Открытый текст не должен быть пустой строкой. Строчные символы автоматически преобразуются к прописным. Все не-буквы удаляются

Исключения

<code>cipher_error</code> ,если	текст пустой
---------------------------------	--------------

4.2.2.3 `getValidCipherText()`

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

Метод для валидации текста зашифрования

.

Исключения

<code>cipher_error</code> ,если	текст пустой или в тексте содержатся строчные буквы
---------------------------------	---

4.2.2.4 `getValidKey()`

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

Проверка на хороший ключ

Исключения

<code>cipher_error</code> ,если	ключ пустой или в ключе находится символ не принадлежащий алфавиту
---------------------------------	--

4.2.2.5 `getValidOpenText()`

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

Метод для валидации открытого текста

Все символы не принадлежащие алфавиту игнорируются

Все строчные символы преобразуются к прописным

Исключения

<code>cipher_error</code> ,если	текст пустой
---------------------------------	--------------

Объявления и описания членов классов находятся в файлах:

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

Глава 5

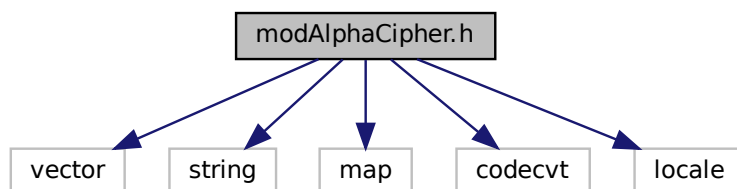
Файлы

5.1 Файл modAlphaCipher.h

Класс-исключение

```
#include <vector>
#include <string>
#include <map>
#include <codecvt>
#include <locale>
```

Граф включаемых заголовочных файлов для modAlphaCipher.h:



Классы

- class `modAlphaCipher`
Шифрование методом Гронсфельда
- class `cipher_error`

5.1.1 Подробное описание

Класс-исключение

Автор

Калгудаева А.Е.

Версия

1.0

Дата

19.12.2023

Авторство

ИБСТ ПГУ

Предупреждения

Это учебный пример

Предметный указатель

`cipher_error`, [7](#)

`decrypt`

`modAlphaCipher`, [9](#)

`encrypt`

`modAlphaCipher`, [9](#)

`getValidCipherText`

`modAlphaCipher`, [10](#)

`getValidKey`

`modAlphaCipher`, [10](#)

`getValidOpenText`

`modAlphaCipher`, [10](#)

`modAlphaCipher`, [8](#)

`decrypt`, [9](#)

`encrypt`, [9](#)

`getValidCipherText`, [10](#)

`getValidKey`, [10](#)

`getValidOpenText`, [10](#)

`modAlphaCipher.h`, [13](#)