Шифр методом Гронсфельда

1.0

Создано системой Doxygen 1.9.1

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

Глава 4

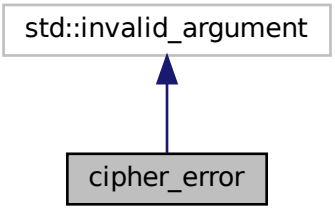
Классы

4.1 Класс cipher_error

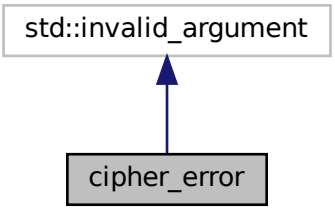
Обработка исключений Класс,созданный для обработки ошибок

#include <modAlpha.h>

Граф наследования:cipher_error:



Граф связей класса cipher_error:



Открытые члены

- cipher_error (const std::string &what_arg)
  Конструктор с параметром типа const std::string.
- cipher_error (const char *what_arg)
  Конструктор с параметром типа const char.

4.1.1 Подробное описание

Обработка исключений Класс,созданный для обработки ошибок

Наследуется от std::invalid_argument

4.1.2 Конструктор(ы)

4.1.2.1 cipher_error() [1/2]

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

Конструктор с параметром типа const std::string.

Аргументы

what_arg	Описание ошибки

4.1.2.2 cipher_error() [2/2]

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

Конструктор с параметром типа const char.

Аргументы

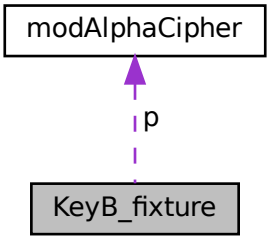
what_arg	Описание ошибки

Объявления и описания членов класса находятся в файле:

- modAlpha.h

4.2 Структура KeyB_fixture

Граф связей класса KeyB_fixture:



Открытые атрибуты

- modAlphaCipher * p

Объявления и описания членов структуры находятся в файле:

- main.cpp

4.3 Класс modAlphaCipher

Шифрование методом Гронсфельда

#include <modAlpha.h>

Открытые члены

- modAlphaCipher ()=delete
  Запрет на использование конструктора по умолчанию
- modAlphaCipher (const std::wstring &skey)
  Конструктор с параметром
- std::wstring encrypt (const std::wstring &open_text)
  Метод для зашифровывания
- std::wstring decrypt (const std::wstring &cipher_text)
  Метод для расшифрования

Закрытые члены

- std::vector< int > convert (const std::wstring &s)

  Преобразование строки в вектор
- std::wstring convert (const std::vector< int > &v)

  Преобразование вектора в строку
- std::wstring getValidKey (const std::wstring &s)

  Валидация ключа
- std::wstring getValidOpenText (const std::wstring &s)

  Валидация открытого текста
- std::wstring getValidCipherText (const std::wstring &s)

  Валидация зашифрованного текста

Закрытые данные

- std::wstring numAlpha = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"

  Алфавит по порядку
- std::map< wchar_t, int > alphaNum

  Ассоциативный массив"номер по символу".
- std::vector< int > key

  Ключ

4.3.1 Подробное описание

Шифрование методом Гронсфельда

Ключ устанавливается в конструкторе. Для зашифровывания и расшифровывания предназначены методы encrypt и decrypt.

Предупреждения

Реализация только для русского языка

4.3.2 Конструктор(ы)

4.3.2.1 modAlphaCipher()

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

Конструктор с параметром

Аргументы

skey	Ключ для шифрования/расшифрования

Исключения

cipher_error	Если ключ невалидный

4.3.3 Методы

4.3.3.1 decrypt()

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

Метод для расшифрования

Аргументы

cipher_text	Зашифрованный текст

Исключения

cipher_error	Если зашифрованный текст невалидный

Возвращает

Расшифрованный текст

4.3.3.2 encrypt()

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

Метод для зашифровывания

Аргументы

open_text	Открытый текст Строчные символы автоматически преобразуются к прописным. Все символы, не являющиеся буквами, удаляются

Исключения

cipher_error	Если открытый текст невалидный

Возвращает

Зашифрованный текст

4.3.3.3 getValidCipherText()

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

Валидация зашифрованного текста

Проверяет, что зашифрованный текст содержит только допустимые символы алфавита.

Аргументы

s	Входная строка, представляющая зашифрованныйтекст.

Исключения

cipher_error	Если зашифрованный текст содержит недопустимые символы.

4.3.3.4 getValidKey()

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

Валидация ключа

Проверяет, что ключ не пустой и не содержит символов, не принадлежащих алфавиту.

Аргументы

s	Входная строка, представляющая ключ.

Исключения

cipher_error	Если ключ пустой или содержит недопустимые символы.

4.3.3.5 getValidOpenText()

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

Валидация открытого текста

Проверяет, что открытый текст содержит только допустимые символы алфавита.

Аргументы

s	Входная строка, представляющая открытый текст.

Исключения

cipher_error	Если открытый текст содержит недопустимые символы.

Объявления и описания членов классов находятся в файлах:

- modAlpha.h
- modAlpha.cpp

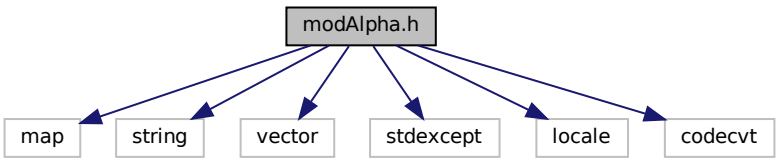
Глава 5

Файлы

5.1 Файл modAlpha.h

Заголовочный файл для модуля Гронсфельда

#include <map>
#include <string>
#include <vector>
#include <stdexcept>
#include <locale>
#include <codecvt>
Граф включаемых заголовочных файлов для modAlpha.h:



Классы

- class modAlphaCipher

  Шифрование методом Гронсфельда
- class cipher_error

  Обработка исключений Класс,созданный для обработки ошибок

5.1.1 Подробное описание

Заголовочный файл для модуля Гронсфельда

Автор

Донскова А.Д

Версия

1.0

Дата

19.12.2024

Авторство

ИБСТ ПГУ

Предупреждения

Это учебный пример

Предметный указатель