modAlphaCipher

1.0

Создано системой Doxygen 1.8.17

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

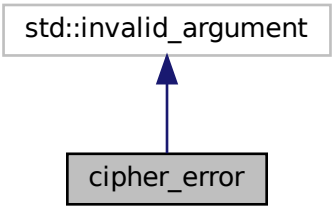
Классы

4.1 Класс cipher_error

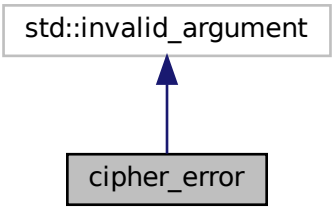
Класс, предназначенный для обработки исключений

#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, предназначенный для обработки исключений

Объявления и описания членов класса находятся в файле:

- modAlphaCipher.h

4.2 Класс modAlphaCipher

Класс, который реализует шифрование методом "Гронсвельда".

#include <modAlphaCipher.h>

Открытые члены

- modAlphaCipher ()=delete
  Запрещённый конструктор без параметров
- modAlphaCipher (const std::wstring &wskey)
  Конструктор для ключа
- std::wstring encrypt (const std::wstring &open_text)
  Метод для шифрования
- std::wstring decrypt (const std::wstring &cipher_text)
  Метод, предназначенный для расшифрования

Закрытые члены

- std::vector< int > convert (const std::wstring &ws)
  Преобразование строки в вектор
- std::wstring convert (const std::vector< int > &v)
  Преобразование вектора в строку
- std::wstring getValidKey (const std::wstring &ws)
  Валидация ключа
- std::wstring getValidOpenText (const std::wstring &ws)
  Валидация текста при шифровании или расшифровании
- std::wstring getValidCipherText (const std::wstring &ws)
  Проверка на правильность текста для зашифровки

Закрытые данные

- std::wstring_convert< std::codecvt_utf8< wchar_t >, wchar_t > codec

  Codec для преобразовния в широкий формат строки и обратно
- std::wstring numAlpha = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"

  Используемый алфавит по порядку для сообщений, которые шифруются методом "Гронсвельда".
- std::map< wchar_t, int > alphaNum

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

  Атрибут, хранящий в себе ключ для шифрования и расшифрования

4.2.1 Подробное описание

Класс, который реализует шифрование методом "Гронсвельда".

Предупреждения

Работает только с русскоязычными сообщениями

4.2.2 Конструктор(ы)

4.2.2.1 modAlphaCipher()

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

Конструктор для ключа

Цикл for построен по строке-алфавиту и на каждом шаге добавляет в ассоциативный массив символ и его номер.

```
for (unsigned i=0; i<numAlpha.size(); i++) {
    alphaNum[numAlpha[i]]=i;
}
```

Аргументы

std::wstring	- ключ в виде строки

4.2.3 Методы

4.2.3.1 convert() [1/2]

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

Преобразование вектора в строку

В переменную типа "wstring" с именем "result" записывается строка согласно индексам каждой буквы алфавита "numAlpha". Индексы хранятся в векторе типа "int", который поступил на вход.

```
wstring result;
 for(auto i:v) {
    result.push_back(numAlpha[i]);
 }
```

Возвращает

строка текста типа "wstring"

4.2.3.2 convert() [2/2]

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

Преобразование строки в вектор

В вектор типа "int" с именем "result" записываются числа, которые являются индексами алфавита "numAlpha",применяемый для строки, которая поступила на вход.

```
vector<int> result;
 for(auto c:s) {
    result.push_back(alphaNum[c]);
 }
```

Возвращает

std::vector <int>, в котором хранятся индексы букв сообщения из алфавита "numAlpha"

4.2.3.3 decrypt()

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

Метод, предназначенный для расшифрования

Здесь сначала формируется вектор work из строки щифратекста с помощью метода convert(). А также происходит проверка шифротекста на наличие ошибки при помощи метода getValidAlphabet↩ Text().

```
vector<int> work = convert(getValidAlphabetText(cipher_text));
```

Если при зашифровывании мы прибавляли значение ключа, то при расшифровывании значения ключа надо вычитать. А чтобы не получить отрицательных значений, выполняется еще прибавление значения модуля, так как такое прибавление не влияет на результат модулю.

```
for(unsigned i=0; i < work.size(); i++) {
      work[i] = (work[i] + alphaNum.size() - key[i % key.size()]) % alphaNum.size();
}
```

Аргументы

std::wstring	cipher_text - сообщение, которое нужно расшифровать

Исключения

cipher_error,если	строка, которая поступила на вход пустая или в ней есть недопустимые символы

Возвращает

строка расшифрованного текста типа "wstring"

4.2.3.4 encrypt()

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

Метод для шифрования

Здесь сначала формируется вектор work из строки открытого текста с помощью метода convert(). А также происходит проверка текста на наличие ошибки при помощи метода getValidAlphabetText().
```
vector<int> work = convert(getValidAlphabetText(open_text));
```

Затем в цикле к каждому элементу вектора прибавляется элемент ключа по модулю размера алфавита. Так как ключ может быть короче текста, то при индексации ключа выполняется операция по модулю размера ключа. Это позволяет использовать ключ циклически на длинных сообщениях.
```
for(unsigned i=0; i < work.size(); i++) {
  work[i] = (work[i] + key[i % key.size()]) % alphaNum.size();
}
```

Далее, при возврате значения, вектор work опять преобразуется в строку.

Аргументы

std::wstring	open_text - сообщение, которое нужно зашифровать

Исключения

cipher_error,если	строка, которая поступила на вход пустая или в ней есть недопустимые символы

Возвращает

строка зашифрованного текста типа "wstring"

4.2.3.5 getValidKey()

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

Валидация ключа

Сначала введённый ключ проверяется на пустоту при помощи обычного условия. Если ключ не пустой, то он проверяется на наличие недопустимых символов.

Предупреждения

Строчные буквы алфавита переводятся в прописные.

Аргументы

std::wstring	s - ключ, который нужно проверить на наличие ошибок, в виде строки

Исключения

cipher_error,если	ключ пустой или в нём присутствуют недопустимые символы

Возвращает

Ключ в виде строки типа "wstring", который успешно прошёл валидацию

4.2.3.6 getValidOpenText()

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

Валидация текста при шифровании или расшифровании

Сначала введённый текст проверяется на пустоту при помощи обычного условия. Если текст не пустой, то он проверяется на наличие недопустимых символов.

Предупреждения

Строчные буквы алфавита переводятся в прописные.

Аргументы

std::wstring	s - строка текста для шифрования или расшифрования, которая проверяется на наличие ошибок

Исключения

cipher_error,если	текст является пустым или в нём присутствуют недопустимые символы

Возвращает

Текст в виде строки типа "wstring", который успешно прошёл валидацию

Объявления и описания членов классов находятся в файлах:

- modAlphaCipher.h
- modAlphaCipher.cpp

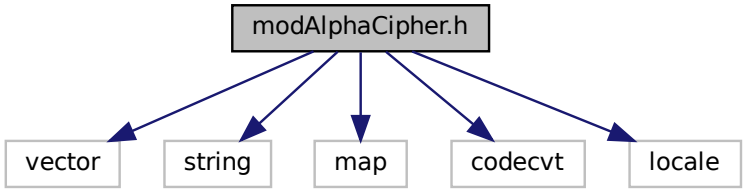
Глава 5

Файлы

5.1 Файл modAlphaCipher.h

Описание класса modAlphaCipher.

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



Классы

- class modAlphaCipher

  Класс, который реализует шифрование методом "Гронсвельда".
- class cipher_error

  Класс, предназначенный для обработки исключений

5.1.1 Подробное описание

Описание класса modAlphaCipher.

Автор

Петров Д.С.

Версия

1.0

Дата

02.06.2021

Авторство

ИБСТ ПГУ

Предметный указатель