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

Шифрование методом табличной перестановки

#include <Cipher.h>

Открытые члены

- Cipher ()=delete
  Конструктор по умолчанию запрещён
- Cipher (std::wstring &ws_key)
  Конструктор принимает ключ (количество столбцов в таблице)
- std::wstring encrypt (std::wstring &ws_open_text)
  Метод для зашифрования
- std::wstring decrypt (const std::wstring &ws_cipher_text)
  Метод для расшифрования
- void set_tableform (const std::wstring &ws_text)
  Формирование информации о таблице
- void set_key (std::wstring &ws_key)
  Установка нового ключа
- int getValidKey (std::wstring &ws_key)
  Проверка на правильность ключа
- std::wstring getValidOpenText (const std::wstring &ws_open_text)
  Проверка на правильность текста для зашифровки
- std::wstring getValidCipherText (const std::wstring &ws_cipher_text)
  Проверка на правильность текста для расшифровки

Закрытые данные

- std::wstring_convert< std::codecvt_utf8< wchar_t >, wchar_t > codec
  codec для преобразования в широкий формат строки и обратно
- int columns
  Количество столбцов в таблице (ключ)
- int rows
  Количество строк в таблице
- int len_text
  Количество символов в слове

4.1.1 Подробное описание

Шифрование методом табличной перестановки

Ключ устанавливается в конструкторе, а также с помощью метода set_key. Для зашифрования и расшифрования предназначены методы encrypt и decrypt. Метод set_tableform - вспомогающий. Методы: getValidKey, getValidOpenText, getValidCipherText - специализируются на проверке входных данных.

Предупреждения

Реализация только для русского языка! С использованием wstring. Шифрование методом табличной перестановки

4.1.2 Конструктор(ы)

4.1.2.1 Cipher()

```
Cipher::Cipher (
            std::wstring & ws_key )
```

Конструктор принимает ключ (количество столбцов в таблице)

Конструктор, принимающий на вход ключ, устанавливает кол-во столбцов

Аргументы

ws_key	

Возвращает

Ничего не возвращает

4.1.3 Методы

4.1.3.1 decrypt()

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

Метод для расшифрования

Метод decrypt расшифровывает текст.

Аргументы

cipher_text	

Возвращает

Зашифрованный текст.

4.1.3.2 encrypt()

std::wstring Cipher::encrypt (
std::wstring & open_text)

Метод для зашифрования

Метод encrypt зашифровывает текст.

Аргументы

open_text	

Возвращает

Зашифрованный текст

4.1.3.3 getValidCipherText()

std::wstring Cipher::getValidCipherText (
const std::wstring & ws_cipher_text) [inline]

Проверка на правильность текста для расшифровки

Данный метод проверяет зашифрованный текст на правильность.

Аргументы

ws_cipher_text	

Возвращает

Зашифрованный текст

Исключения

cipher_error,если	текст пустой или невалидный

4.1.3.4 getValidKey()

int Cipher::getValidKey (
std::wstring & ws_key) [inline]

Проверка на правильность ключа

Данный метод проверяет ключ на правильность.

Аргументы

ws_key	

Возвращает

Ключ

Исключения

cipher_error,если	ключ пустой или невалидный

4.1.3.5 getValidOpenText()

std::wstring Cipher::getValidOpenText (
const std::wstring & ws_open_text) [inline]

Проверка на правильность текста для зашифровки

Данный метод проверяет открытый текст на правильность. Строчные буквы превращаются в прописные. Все не-буквы удаляются.

Аргументы

ws_open_text	

Возвращает

Текст для расшифровки

Исключения

cipher_error,если	текст пустой

4.1.3.6 set_key()

void Cipher::set_key (
std::wstring & ws_key)

Установка нового ключа

Метод, принимающий на вход ключ, устанавливает кол-во столбцов

Аргументы

ws_key	

Возвращает

Ничего не возвращает

4.1.3.7 set_tableform()

void Cipher::set_tableform (
const std::wstring & open_text)

Формирование информации о таблице

Принимает текст для зашифровки, далее по нему формирует кол-во строк в таблице, а также получает длину текста.

Аргументы

open_text	

Объявления и описания членов классов находятся в файлах:

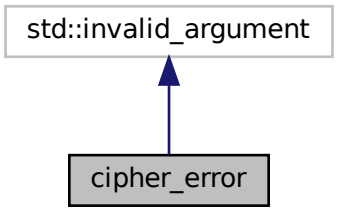
- Cipher.h
- Cipher.cpp

4.2 Класс cipher_error

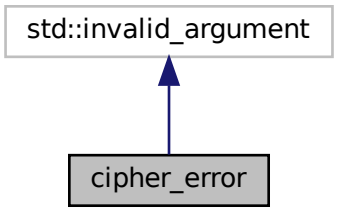
Класс-исключение

#include <Cipher.h>

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 си строку, поднимает исключение

4.2.1 Подробное описание

Класс-исключение

4.2.2 Конструктор(ы)

4.2.2.1 cipher_error() [1/2]

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

Принимает строку, поднимает исключение

Аргументы

what_arg	

4.2.2.2 cipher_error() [2/2]

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

Принимает си строку, поднимает исключение

Аргументы

what_arg	

Объявления и описания членов класса находятся в файле:

- Cipher.h

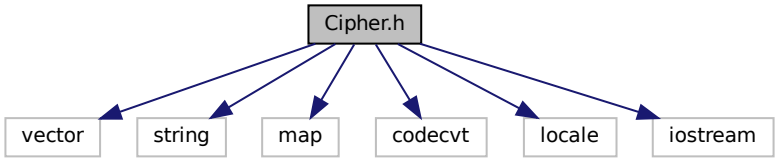
Глава 5

Файлы

5.1 Файл Cipher.h

Класс-исключение

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



Классы

- class Cipher

  Шифрование методом табличной перестановки
- class cipher_error

  Класс-исключение

5.1.1 Подробное описание

Класс-исключение

Предметный указатель