

Шифрование методом табличной маршрутной перестановки

Создано системой Doxygen 1.9.4

Глава 1

Алфавитный указатель классов

1.1 Классы

Классы с их кратким описанием.

`modPermutationCipher`

Класс для реализации шифра перестановки с ключом ??

Глава 2

Список файлов

2.1 Файлы

Полный список документированных файлов.

[modPermutation.h](#) ??

Глава 3

Классы

3.1 Класс modPermutationCipher

Класс для реализации шифра перестановки с ключом.

```
#include <modPermutation.h>
```

Открытые члены

- `modPermutationCipher` (const std::wstring &key)
Конструктор класса.
- std::wstring `encrypt` (const std::wstring &open_text)
Метод для шифрования текста.
- std::wstring `decrypt` (const std::wstring &cipher_text)
Метод для расшифрования текста.
- void `validateKey` (const std::wstring &key)
Проверяет корректность ключа.
- void `validateText` (const std::wstring &text)
Проверяет корректность текста.

Закрытые данные

- std::wstring alphabet
Алфавит, используемый для шифрования (русские и английские буквы).
- std::vector< int > key
Ключ шифрования в виде вектора целых чисел.

3.1.1 Подробное описание

Класс для реализации шифра перестановки с ключом.

Этот класс предоставляет функциональность для шифрования и расшифрования текста на основе алгоритма перестановки с использованием числового ключа.

3.1.2 Конструктор(ы)

3.1.2.1 modPermutationCipher()

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

Конструктор класса.

Конструктор класса [modPermutationCipher](#).

Инициализирует объект с заданным ключом.

Аргументы

skey	Ключ для шифрования в формате строки.
------	---------------------------------------

Исключения

std::invalid_argument	Если ключ некорректен.
-----------------------	------------------------

Конструктор инициализирует объект с переданным ключом. Также выполняется проверка ключа на корректность: он должен быть непустым и содержать только цифры.

Аргументы

skey	Ключ в виде строки, состоящей из цифр.
------	--

Исключения

std::invalid_argument	Исключение выбрасывается, если ключ пуст или содержит недопустимые символы.
-----------------------	---

3.1.3 Методы

3.1.3.1 decrypt()

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

Метод для расшифрования текста.

Функция для расшифрования текста.

Аргументы

cipher_text	Шифрованный текст для расшифрования.
-------------	--------------------------------------

Возвращает

std::wstring Расшифрованный текст.

Расшифровывает переданный текст, используя алгоритм перестановки с ключом.

Аргументы

cipher_text	Текст для расшифрования.
-------------	--------------------------

Возвращает

std::wstring Расшифрованный текст.

3.1.3.2 encrypt()

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

Метод для шифрования текста.

Функция для шифрования текста.

Аргументы

open_text	Открытый текст для шифрования.
-----------	--------------------------------

Возвращает

std::wstring Зашифрованный текст.

Шифрует переданный текст, используя алгоритм перестановки с ключом.

Аргументы

open_text	Текст для шифрования.
-----------	-----------------------

Возвращает

std::wstring Зашифрованный текст.

3.1.3.3 validateKey()

```
void modPermutationCipher::validateKey (
    const std::wstring & skey )
```

Проверяет корректность ключа.

Функция для валидации ключа.

Аргументы

key	Ключ для проверки.
-----	--------------------

Исключения

std::invalid_argument	Если ключ некорректен.
-----------------------	------------------------

Проверяет, что ключ состоит только из цифр и является положительным целым числом.

Аргументы

skey	Ключ в виде строки.
------	---------------------

Исключения

std::invalid_argument	Исключение выбрасывается, если ключ содержит нецифровые символы или является неположительным.
-----------------------	---

3.1.3.4 validateText()

```
void modPermutationCipher::validateText (
    const std::wstring & text )
```

Проверяет корректность текста.

Функция для валидации текста.

Аргументы

text	Текст для проверки.
------	---------------------

Исключения

std::invalid_argument	Если текст некорректен.
-----------------------	-------------------------

Проверяет, что текст не пуст и состоит только из символов заданного алфавита (русские и англий-

ские буквы).

Аргументы

text	Текст для шифрования или расшифрования.
------	---

Исключения

std::invalid_argument	Исключение выбрасывается, если текст содержит недопустимые символы.
-----------------------	---

Объявления и описания членов классов находятся в файлах:

- modPermutation.h
- modPermutation.cpp

Глава 4

Файлы

4.1 modPermutation.h

```
1 #pragma once
2 #include <string>
3 #include <vector>
4 #include <stdexcept>
5 #include <locale>
6 #include <map>
7 #include <algorithm>
8 #include <iostream>
9
10 class modPermutationCipher {
11 private:
12     std::wstring alphabet;
13     std::vector<int> key;
14
15 public:
16     modPermutationCipher(const std::wstring& skey);
17
18     std::wstring encrypt(const std::wstring& open_text);
19
20     std::wstring decrypt(const std::wstring& cipher_text);
21
22     void validateKey(const std::wstring& key);
23
24     void validateText(const std::wstring& text);
25 };
```

