

Zabolotskikh23ptlab4

Создано системой Doxygen 1.9.4

1 Алфавитный указатель классов	1
1.1 Классы	1
2 Список файлов	3
2.1 Файлы	3
3 Классы	5
3.1 Класс <code>modAlphaCipher</code>	5
3.1.1 Подробное описание	6
3.1.2 Конструктор(ы)	6
3.1.2.1 <code>modAlphaCipher()</code> [1/2]	6
3.1.2.2 <code>modAlphaCipher()</code> [2/2]	6
3.1.3 Методы	6
3.1.3.1 <code>convert()</code> [1/2]	6
3.1.3.2 <code>convert()</code> [2/2]	7
3.1.3.3 <code>decrypt()</code>	7
3.1.3.4 <code>encrypt()</code>	8
3.1.3.5 <code>isValid()</code>	8
4 Файлы	9
4.1 Файл <code>modAlphaCipher.h</code>	9
4.1.1 Подробное описание	9
4.2 <code>modAlphaCipher.h</code>	10
Предметный указатель	11

Глава 1

Алфавитный указатель классов

1.1 Классы

Классы с их кратким описанием.

[modAlphaCipher](#)

Класс для шифрования и дешифрования текста с использованием модульного алфавитного шифра

[5](#)

Глава 2

Список файлов

2.1 Файлы

Полный список документированных файлов.

modAlphaCipher.h	
Описание класса modAlphaCipher	9

Глава 3

Классы

3.1 Класс modAlphaCipher

Класс для шифрования и дешифрования текста с использованием модульного алфавитного шифра.

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

Открытые члены

- `modAlphaCipher ()=delete`
Конструктор по умолчанию удален.
- `modAlphaCipher (const wstring &skey)`
Конструктор класса `modAlphaCipher`.
- `wstring encrypt (const wstring &open_text)`
Шифрует открытый текст с использованием заданного ключа.
- `wstring decrypt (const wstring &cipher_text)`
Дешифрует зашифрованный текст с использованием заданного ключа.

Закрытые члены

- `vector< int > convert (const wstring &s)`
Преобразует строку в вектор целых чисел на основе индексов символов.
- `wstring convert (const vector< int > &v)`
Преобразует вектор целых чисел обратно в строку.
- `bool isValid (const wstring &s)`
Проверяет, является ли строка допустимой для шифрования.

Закрытые данные

- `wstring numAlpha = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"`
Кириллический алфавит.
- `map< wchar_t, int > alphaNum`
Сопоставление символов алфавита с их индексами.
- `vector< int > key`
Вектор ключа для шифрования/дешифрования.

3.1.1 Подробное описание

Класс для шифрования и дешифрования текста с использованием модульного алфавитного шифра.

Этот класс реализует модульный алфавитный шифр, который позволяет шифровать и дешифровать текст на основе заданного ключа, используя кириллический алфавит.

3.1.2 Конструктор(ы)

3.1.2.1 modAlphaCipher() [1/2]

```
modAlphaCipher::modAlphaCipher ( ) [delete]
```

Конструктор по умолчанию удален.

Запрещает создание объекта без ключа.

3.1.2.2 modAlphaCipher() [2/2]

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

Конструктор класса [modAlphaCipher](#).

Аргументы

skey	Строка-ключ для шифрования/дешифрования. Должна содержать только допустимые символы.
------	--

Исключения

invalid_argument	Если ключ содержит недопустимые символы.
------------------	--

3.1.3 Методы

3.1.3.1 convert() [1/2]

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

Преобразует вектор целых чисел обратно в строку.

Аргументы

v	Вектор целых чисел для преобразования.
---	--

Возвращает

Строка, представляющая символы по индексам.

3.1.3.2 convert() [2/2]

```
vector< int > modAlphaCipher::convert (
    const wstring & s ) [private]
```

Преобразует строку в вектор целых чисел на основе индексов символов.

Аргументы

s	Строка для преобразования.
---	----------------------------

Возвращает

Вектор целых чисел, представляющий индексы символов.

3.1.3.3 decrypt()

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

Дешифрует зашифрованный текст с использованием заданного ключа.

Аргументы

cipher_text	Зашифрованный текст, который нужно расшифровать.
-------------	--

Возвращает

Открытый текст.

Исключения

invalid_argument	Если зашифрованный текст содержит недопустимые символы.
------------------	---

3.1.3.4 encrypt()

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

Шифрует открытый текст с использованием заданного ключа.

Аргументы

open_text	Открытый текст, который нужно зашифровать.
-----------	--

Возвращает

Зашифрованный текст.

Исключения

invalid_argument	Если открытый текст содержит недопустимые символы.
------------------	--

3.1.3.5 isValid()

```
bool modAlphaCipher::isValid (  
    const wstring & s ) [private]
```

Проверяет, является ли строка допустимой для шифрования.

Аргументы

s	Строка для проверки.
---	----------------------

Возвращает

true, если строка допустима; false в противном случае.

Объявления и описания членов класса находятся в файле:

- [modAlphaCipher.h](#)

Глава 4

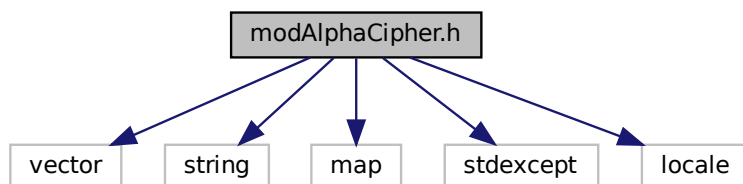
Файлы

4.1 Файл modAlphaCipher.h

Описание класса [modAlphaCipher](#).

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

Граф включаемых заголовочных файлов для modAlphaCipher.h:



Классы

- class [modAlphaCipher](#)

Класс для шифрования и дешифрования текста с использованием модульного алфавитного шифра.

4.1.1 Подробное описание

Описание класса [modAlphaCipher](#).

Автор

Заболотских З.В.

Версия

1.0

Дата

29.11.2024

4.2 modAlphaCipher.h

[См. документацию.](#)

```
1
2
3 #pragma once
4 #include <vector>
5 #include <string>
6 #include <map>
7 #include <stdexcept>
8 #include <locale>
9
10 using namespace std;
11
12 class modAlphaCipher {
13 private:
14     wstring numAlpha = L"АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ";
15     map<wchar_t, int> alphaNum;
16     vector<int> key;
17
18     vector<int> convert(const wstring& s);
19
20     wstring convert(const vector<int>& v);
21
22     bool isValid(const wstring &s); // Объявление функции проверки
23
24 public:
25     modAlphaCipher() = delete;
26
27     modAlphaCipher(const wstring& skey);
28
29     wstring encrypt(const wstring& open_text);
30
31     wstring decrypt(const wstring& cipher_text);
32 };
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```

Предметный указатель

convert
 modAlphaCipher, [6](#), [7](#)

decrypt
 modAlphaCipher, [7](#)

encrypt
 modAlphaCipher, [8](#)

isValid
 modAlphaCipher, [8](#)

modAlphaCipher, [5](#)
 convert, [6](#), [7](#)
 decrypt, [7](#)
 encrypt, [8](#)
 isValid, [8](#)
 modAlphaCipher, [6](#)
modAlphaCipher.h, [9](#)