

Шифр маршрутной перестановки

Создано системой Doxygen 1.9.1

1 Иерархический список классов	1
1.1 Иерархия классов	1
2 Алфавитный указатель классов	3
2.1 Классы	3
3 Список файлов	5
3.1 Файлы	5
4 Классы	7
4.1 Класс cipher_error	7
4.1.1 Подробное описание	8
4.1.2 Конструктор(ы)	8
4.1.2.1 cipher_error() [1/2]	8
4.1.2.2 cipher_error() [2/2]	8
4.2 Структура Key3Fixture	9
4.2.1 Подробное описание	9
4.2.2 Конструктор(ы)	9
4.2.2.1 Key3Fixture()	9
4.2.2.2 ~Key3Fixture()	10
4.2.3 Данные класса	10
4.2.3.1 cipher	10
4.3 Класс Table	10
4.3.1 Подробное описание	10
4.3.2 Конструктор(ы)	10
4.3.2.1 Table()	10
4.3.3 Методы	11
4.3.3.1 decrypt()	11
4.3.3.2 encrypt()	12
5 Файлы	13
5.1 Файл Desktop/4лб/task2/main.cpp	13
5.1.1 Подробное описание	14
5.1.2 Функции	14
5.1.2.1 main()	14
5.1.2.2 str8_to_w()	14
5.1.2.3 w_to_str8()	15
5.2 Файл Desktop/4лб/task2/table.cpp	15
5.2.1 Подробное описание	16
5.3 Файл Desktop/4лб/task2/table.h	16
5.3.1 Подробное описание	17
5.4 Файл Desktop/4лб/task2/test_table.cpp	17
5.4.1 Подробное описание	18
5.4.2 Макросы	18
5.4.2.1 CHECK_WIDE_EQUAL	19

5.4.3 Функции	19
5.4.3.1 main()	19
5.4.3.2 SUITE() [1/3]	19
5.4.3.3 SUITE() [2/3]	19
5.4.3.4 SUITE() [3/3]	20
5.4.3.5 utf8ToWide()	20
5.4.3.6 wideToUtf8()	20
Предметный указатель	21

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

invalid_argument	
cipher_error	7
Key3Fixture	9
Table	10

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

cipher_error	Класс-исключение для ошибок шифрования	7
Key3Fixture	Фикстура для тестов с ключом 3	9
Table	Класс для шифрования табличной маршрутной перестановкой	10

Глава 3

Список файлов

3.1 Файлы

Полный список файлов.

Desktop/4лб/task2/ main.cpp	
Главный модуль программы для шифрования/расшифрования табличной перестановкой	13
Desktop/4лб/task2/ table.cpp	
Реализация методов класса Table для табличной маршрутной перестановки . . .	15
Desktop/4лб/task2/ table.h	
Заголовочный файл для модуля табличной маршрутной перестановки	16
Desktop/4лб/task2/ test_table.cpp	
Модульные тесты для класса Table	17

Глава 4

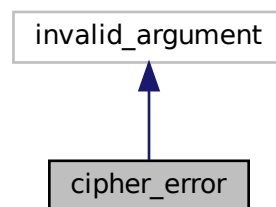
Классы

4.1 Класс `cipher_error`

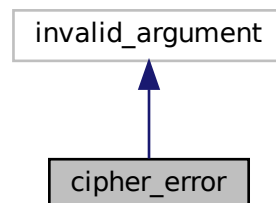
Класс-исключение для ошибок шифрования

```
#include <table.h>
```

Граф наследования: `cipher_error`:



Граф связей класса `cipher_error`:



Открытые члены

- `cipher_error` (const string &what_arg)
Конструктор с параметром string.
- `cipher_error` (const char *what_arg)
Конструктор с параметром const char*.

4.1.1 Подробное описание

Класс-исключение для ошибок шифрования

Наследуется от `std::invalid_argument`

4.1.2 Конструктор(ы)

4.1.2.1 `cipher_error()` [1/2]

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

Конструктор с параметром string.

Аргументы

what_arg	сообщение об ошибке
----------	---------------------

4.1.2.2 `cipher_error()` [2/2]

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

Конструктор с параметром const char*.

Аргументы

what_arg	сообщение об ошибке
----------	---------------------

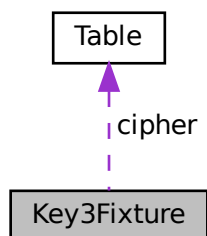
Объявления и описания членов класса находятся в файле:

- Desktop/4лб/task2/[table.h](#)

4.2 Структура Key3Fixture

Фикстура для тестов с ключом 3.

Граф связей класса Key3Fixture:



Открытые члены

- [Key3Fixture \(\)](#)
- [~Key3Fixture \(\)](#)

Открытые атрибуты

- [Table * cipher](#)

4.2.1 Подробное описание

Фикстура для тестов с ключом 3.

Создает экземпляр шифратора с 3 столбцами для повторного использования

4.2.2 Конструктор(ы)

4.2.2.1 Key3Fixture()

`Key3Fixture::Key3Fixture ()` [inline]

4.2.2.2 ~Key3Fixture()

Key3Fixture::~~Key3Fixture () [inline]

4.2.3 Данные класса

4.2.3.1 cipher

[Table*](#) Key3Fixture::cipher

Объявления и описания членов структуры находятся в файле:

- Desktop/4лб/task2/[test_table.cpp](#)

4.3 Класс Table

Класс для шифрования табличной маршрутной перестановкой

```
#include <table.h>
```

Открытые члены

- [Table](#) (int key)
Конструктор с установкой ключа
- wstring [encrypt](#) (const wstring &plain)
Шифрование открытого текста
- wstring [decrypt](#) (const wstring &cipher)
Расшифрование зашифрованного текста

4.3.1 Подробное описание

Класс для шифрования табличной маршрутной перестановкой

Реализует шифрование и расшифрование текста методом табличной перестановки с маршрутом записи: по горизонтали слева направо, сверху вниз с маршрутом считывания: сверху вниз, справа налево

4.3.2 Конструктор(ы)

4.3.2.1 Table()

```
Table::Table (
    int key ) [explicit]
```

Конструктор с установкой ключа

Конструктор класса [Table](#).

Аргументы

key	количество столбцов таблицы
-----	-----------------------------

Исключения

cipher_error	если ключ невалиден
------------------------------	---------------------

4.3.3 Методы

4.3.3.1 decrypt()

```
wstring Table::decrypt (
    const wstring & cipher )
```

Расшифрование зашифрованного текста

Расшифрование зашифрованного текста табличной перестановкой

Аргументы

cipher	зашифрованный текст для расшифрования
--------	---------------------------------------

Возвращает

расшифрованный текст

Исключения

cipher_error	если текст невалиден
------------------------------	----------------------

Аргументы

cipher	зашифрованный текст для расшифрования
--------	---------------------------------------

Возвращает

расшифрованный текст

Обратный процесс шифрованию с учетом маршрутов

Исключения

cipher_error	если зашифрованный текст невалиден
------------------------------	------------------------------------

4.3.3.2 encrypt()

```
wstring Table::encrypt (  
    const wstring & plain )
```

Шифрование открытого текста

Шифрование открытого текста табличной перестановкой

Аргументы

plain	открытый текст для шифрования
-------	-------------------------------

Возвращает

зашифрованный текст

Исключения

cipher_error	если текст невалиден
------------------------------	----------------------

Аргументы

plain	открытый текст для шифрования
-------	-------------------------------

Возвращает

зашифрованный текст

Маршрут записи: по горизонтали слева направо, сверху вниз

Маршрут считывания: сверху вниз, справа налево

Исключения

cipher_error	если открытый текст невалиден
------------------------------	-------------------------------

Объявления и описания членов классов находятся в файлах:

- Desktop/4лб/task2/[table.h](#)
- Desktop/4лб/task2/[table.cpp](#)

Глава 5

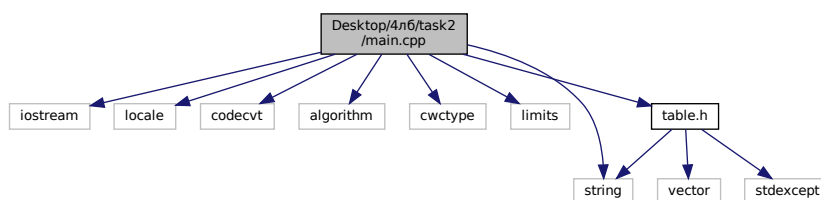
Файлы

5.1 Файл Desktop/4лб/task2/main.cpp

Главный модуль программы для шифрования/расшифрования табличной перестановкой

```
#include <iostream>
#include <locale>
#include <codecvt>
#include <algorithm>
#include <cwctype>
#include <limits>
#include <string>
#include "table.h"
```

Граф включаемых заголовочных файлов для main.cpp:



Функции

- `wstring str8_to_w` (const string &s)
Преобразование строки UTF-8 в широкую строку
- `string w_to_str8` (const wstring &ws)
Преобразование широкой строки в строку UTF-8.
- `int main` ()
Главная функция программы

5.1.1 Подробное описание

Главный модуль программы для шифрования/расшифрования табличной перестановкой

Автор

Соловьев Арсений Евгеньевич

Версия

1.0

Дата

2025

Реализует пользовательский интерфейс для работы с шифром табличной маршрутной перестановки

5.1.2 Функции

5.1.2.1 main()

```
int main ( )
```

Главная функция программы

Возвращает

код завершения программы

Реализует диалоговый интерфейс для шифрования/расшифрования табличной перестановкой

5.1.2.2 str8_to_w()

```
wstring str8_to_w (
    const string & s )
```

Преобразование строки UTF-8 в широкую строку

Аргументы

s	строка в UTF-8
---	----------------

Возвращает

широкая строка

5.1.2.3 w_to_str8()

```
string w_to_str8 (  
    const wstring & ws )
```

Преобразование широкой строки в строку UTF-8.

Аргументы

ws	широкая строка
----	----------------

Возвращает

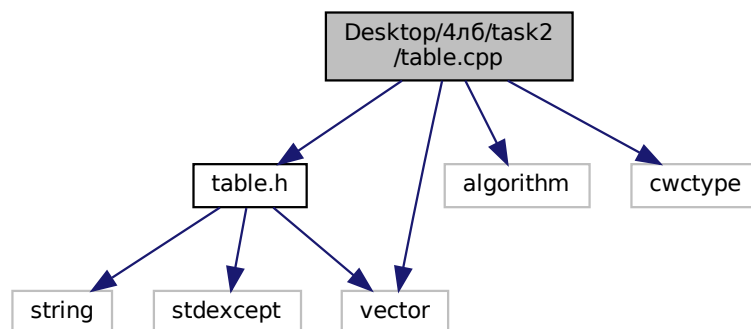
строка в UTF-8

5.2 Файл Desktop/4лб/task2/table.cpp

Реализация методов класса [Table](#) для табличной маршрутной перестановки

```
#include "table.h"  
#include <algorithm>  
#include <vector>  
#include <cwctype>
```

Граф включаемых заголовочных файлов для table.cpp:



5.2.1 Подробное описание

Реализация методов класса [Table](#) для табличной маршрутной перестановки

Автор

Соловьев Арсений Евгеньевич

Версия

1.0

Дата

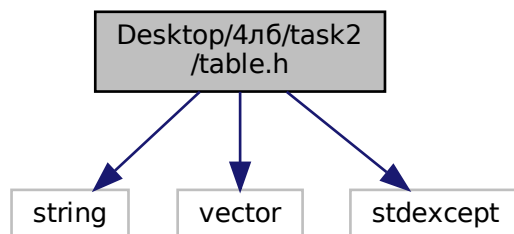
2025

5.3 Файл Desktop/4лб/task2/table.h

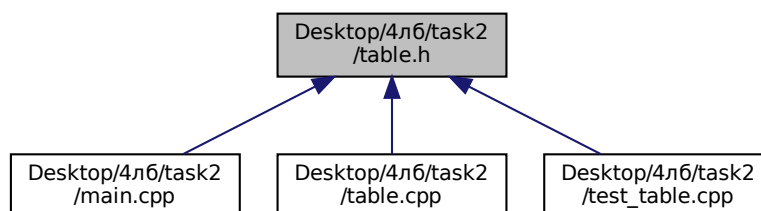
Заголовочный файл для модуля табличной маршрутной перестановки

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

Граф включаемых заголовочных файлов для table.h:



Граф файлов, в которые включается этот файл:



Классы

- class `cipher_error`
Класс-исключение для ошибок шифрования
- class `Table`
Класс для шифрования табличной маршрутной перестановкой

5.3.1 Подробное описание

Заголовочный файл для модуля табличной маршрутной перестановки

Автор

Соловьев Арсений Евгеньевич

Версия

1.0

Дата

2025

Предупреждения

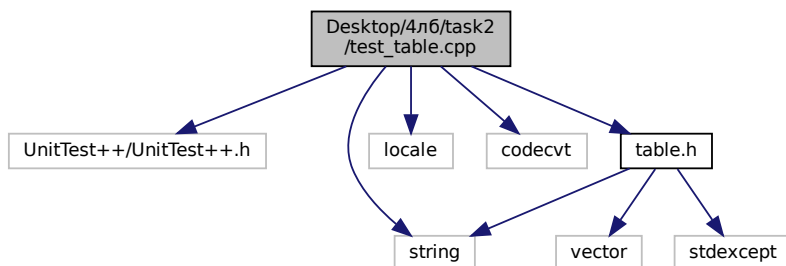
Реализация для русского языка

5.4 Файл Desktop/4лб/task2/test_table.cpp

Модульные тесты для класса `Table`.

```
#include <UnitTest++/UnitTest++.h>
#include <string>
#include <locale>
#include <codecvt>
#include "table.h"
```

Граф включаемых заголовочных файлов для test_table.cpp:



Классы

- struct [Key3Fixture](#)
Фикстура для тестов с ключом 3.

Макросы

- `#define CHECK_WIDE_EQUAL(expected, actual) CHECK_EQUAL(wideToUtf8(expected), wideToUtf8(actual))`
Макрос для сравнения широких строк в тестах

Функции

- string [wideToUtf8](#) (const wstring &ws)
Преобразование широкой строки в UTF-8.
- wstring [utf8ToWide](#) (const string &s)
Преобразование строки UTF-8 в широкую строку
- [SUITE](#) (ConstructorTest)
Тестовый набор для конструктора [Table](#).
- [SUITE](#) (EncryptTest)
Тестовый набор для метода encrypt.
- [SUITE](#) (DecryptTest)
Тестовый набор для метода decrypt.
- int [main](#) (int argc, char **argv)
Главная функция тестов

5.4.1 Подробное описание

Модульные тесты для класса [Table](#).

Автор

Соловьев Арсений Евгеньевич

Версия

1.0

Дата

2025

Тестирование функциональности шифрования табличной перестановкой и обработки ошибок

5.4.2 Макросы

5.4.2.1 CHECK_WIDE_EQUAL

```
#define CHECK_WIDE_EQUAL(  
    expected,  
    actual ) CHECK_EQUAL(wideToUtf8(expected), wideToUtf8(actual))
```

Макрос для сравнения широких строк в тестах

5.4.3 Функции

5.4.3.1 main()

```
int main (  
    int argc,  
    char ** argv )
```

Главная функция тестов

Аргументы

argc	количество аргументов
argv	массив аргументов

Возвращает

результат выполнения тестов

5.4.3.2 SUITE() [1/3]

```
SUITE (  
    ConstructorTest )
```

Тестовый набор для конструктора [Table](#).

Проверяет различные сценарии инициализации ключа (количества столбцов)

5.4.3.3 SUITE() [2/3]

```
SUITE (  
    DecryptTest )
```

Тестовый набор для метода decrypt.

Проверяет различные сценарии расшифрования табличной перестановкой

5.4.3.4 SUITE() [3/3]

```
SUITE (
    EncryptTest )
```

Тестовый набор для метода encrypt.

Проверяет различные сценарии шифрования табличной перестановкой

5.4.3.5 utf8ToWide()

```
wstring utf8ToWide (
    const string & s )
```

Преобразование строки UTF-8 в широкую строку

Аргументы

s	строка в UTF-8
---	----------------

Возвращает

широкая строка

5.4.3.6 wideToUtf8()

```
string wideToUtf8 (
    const wstring & ws )
```

Преобразование широкой строки в UTF-8.

Аргументы

ws	широкая строка
----	----------------

Возвращает

строка в UTF-8

Предметный указатель

- [~Key3Fixture](#)
 - [Key3Fixture](#), [9](#)
- [CHECK_WIDE_EQUAL](#)
 - [test_table.cpp](#), [18](#)
- [cipher](#)
 - [Key3Fixture](#), [10](#)
- [cipher_error](#), [7](#)
 - [cipher_error](#), [8](#)
- [decrypt](#)
 - [Table](#), [11](#)
- [Desktop/4л6/task2/main.cpp](#), [13](#)
- [Desktop/4л6/task2/table.cpp](#), [15](#)
- [Desktop/4л6/task2/table.h](#), [16](#)
- [Desktop/4л6/task2/test_table.cpp](#), [17](#)
- [encrypt](#)
 - [Table](#), [12](#)
- [Key3Fixture](#), [9](#)
 - [~Key3Fixture](#), [9](#)
 - [cipher](#), [10](#)
 - [Key3Fixture](#), [9](#)
- [main](#)
 - [main.cpp](#), [14](#)
 - [test_table.cpp](#), [19](#)
- [main.cpp](#)
 - [main](#), [14](#)
 - [str8_to_w](#), [14](#)
 - [w_to_str8](#), [15](#)
- [str8_to_w](#)
 - [main.cpp](#), [14](#)
- [SUITE](#)
 - [test_table.cpp](#), [19](#)
- [Table](#), [10](#)
 - [decrypt](#), [11](#)
 - [encrypt](#), [12](#)
 - [Table](#), [10](#)
- [test_table.cpp](#)
 - [CHECK_WIDE_EQUAL](#), [18](#)
 - [main](#), [19](#)
 - [SUITE](#), [19](#)
 - [utf8ToWide](#), [20](#)
 - [wideToUtf8](#), [20](#)
- [utf8ToWide](#)
 - [test_table.cpp](#), [20](#)
- [w_to_str8](#)
 - [main.cpp](#), [15](#)
- [wideToUtf8](#)
 - [test_table.cpp](#), [20](#)