

Практическое задание №5

Создать решение с именем **Task5**. Рекомендации см. в файле «Материалы к Практическому заданию №5.pdf».

Необходимо реализовать DLL-библиотеку со следующими функциями:

1. **Функция шифрования** текстового файла, которой передаются имя файла-источника с открытым текстом (`const char*`), имя файла-приёмника с получаемым зашифрованным текстом (`const char*`) и ключ шифрования в виде C-строки (`const char*`). Если файл-источник существует, то функция копирует его содержимое в файл-приёмник, шифруя файл с помощью поразрядной операции «Исключающее ИЛИ, XOR» (поразрядная операция \wedge). Если файл-приёмник не существует, он должен создаваться. Если файл-приёмник существует, то он должен перезаписываться. Особенности шифрования с помощью операции «Исключающее ИЛИ»:

$$[\text{Шифр_текст}] = [\text{Исх_текст}] \wedge [\text{Ключ}]$$

Для шифрования файла должны использоваться циклически все байты ключа (то есть если длина ключа 3 байта, то первый байт файла шифруется первым байтом ключа, второй байт – вторым, третий – третьим, четвёртый – первым байтом ключа и т.д.).

2. **Функция дешифрования** текстового файла передаются имя файла-источника с зашифрованным текстом (`const char*`), имя файла-приёмника с получаемым открытым текстом (`const char*`) и ключ дешифрования в виде C-строки (`const char*`). Если файл-источник существует, то программа копирует его содержимое в файл-приёмник, дешифруя файл с помощью поразрядной операции «Исключающее ИЛИ, XOR» (поразрядная операция \wedge). Если файл-приёмник не существует, он должен создаваться. Если файл-приёмник существует, то он должен перезаписываться. Особенности дешифрования с помощью операции «Исключающее ИЛИ»:

$$[\text{Исх_текст}] = [\text{Шифр_текст}] \wedge [\text{Ключ}]$$

3. **Функция преобразования C-строки** с числом в заданной системе счисления в число типа **double**. Разделителем целой и дробной частей является точка. Дробная часть может отсутствовать (при этом может отсутствовать и точка).

Примеры строк (для 16-чной системы счисления):

“0d97.17a”, “-301.”, “+b74”, “0.0”.

Библиотечные функции не использовать.

C-строка и основание системы счисления (не более 36) передаются в функцию в качестве параметров.

Реализовать **тесты** для проверки разработанных функций. В тестах функции преобразования использовать основание системы счисления, равное (11 + номер слушателя по журналу).

Для сдачи проекта использовать структуру в файловой системе:

gxxxxx/5/Task5/** – файлы с реализацией пунктов задания (имена файлов давать в соответствии с вариантами);

gxxxxx/5/Task5.Tests/** – файлы с тестами пунктов задания (имена файлов давать в соответствии с вариантами).

Необходимые условия сдачи:

- стиль программирования должен соответствовать установленным правилам;
- имена переменных и функций должны быть осмысленными; никакой транслитерации и нелогичности;
- программа должна быть протестирована с помощью gtest;
- программа должна проходить тесты преподавателя (которых вы не видите).