**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра информационных систем**

отчет

**по практической работе №4**

**по дисциплине «Программирование»**

Тема: «Строки»

|  |  |  |
| --- | --- | --- |
| Студент гр. 2373 | Шаймухаметов Р.Н. |  |
| Преподаватель | Глущенко А. Г. |  |

Санкт-Петербург

2022

**Цель работы.**

Изучить строки, способы применения. Научиться работать со строками.

**Основные теоретические положения.**

Текстовые строки представляются с помощью одномерных массивов символов. В языке C++ текстовая строка представляет собой набор символов, обязательно заканчивающийся нулевым символом (‘\0’). Поэтому, если вы хотите создать текстовый массив для хранения 10 (N) символов, нужно выделить память под 11(N+1) символов.

Широкое применение в программировании получила специальная форма двумерных массивов символов, которые можно рассматривать как одномерный массив строк, заканчивающихся нулевым символом.

Класс string предназначен для работы со строками типа char, которые представляют собой строчку с завершающим нулем (символ ‘\0’). Класс string был введен как альтернативный вариант для работы со строками типа char.

При работе со строками часто будет возникать потребность в поиске набора символа или слов (поиска подстроки в строке). При условии, что текст может быть крайне большим, хочется, чтобы алгоритм поиска подстроки работал быстро.

**Постановка задачи.**

Необходимо написать программу, которая:

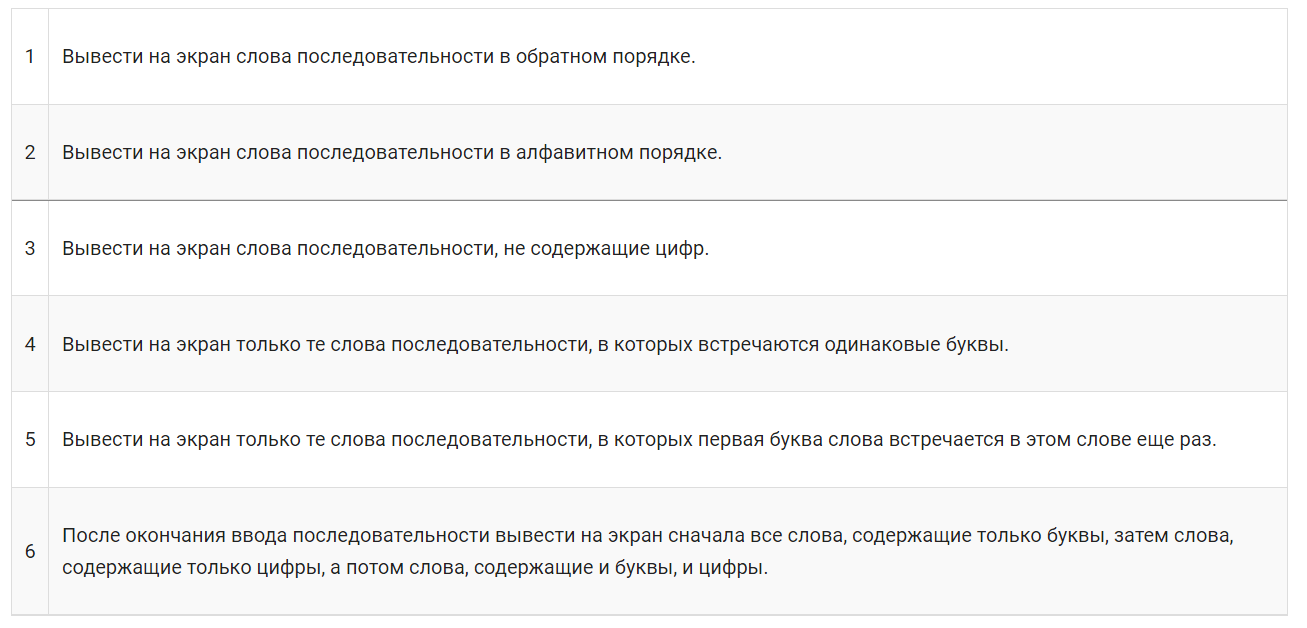
1. С клавиатуры или с файла (\*) (пользователь сам может выбрать способ ввода) вводится последовательность, содержащая от 1 до 50 слов, в каждом из которых от 1 до 10 строчных латинских букв и цифр. Между соседними словами произвольное количество пробелов. За последним символом стоит точка.
2. Необходимо отредактировать входной текст:

·        удалить лишние пробелы;

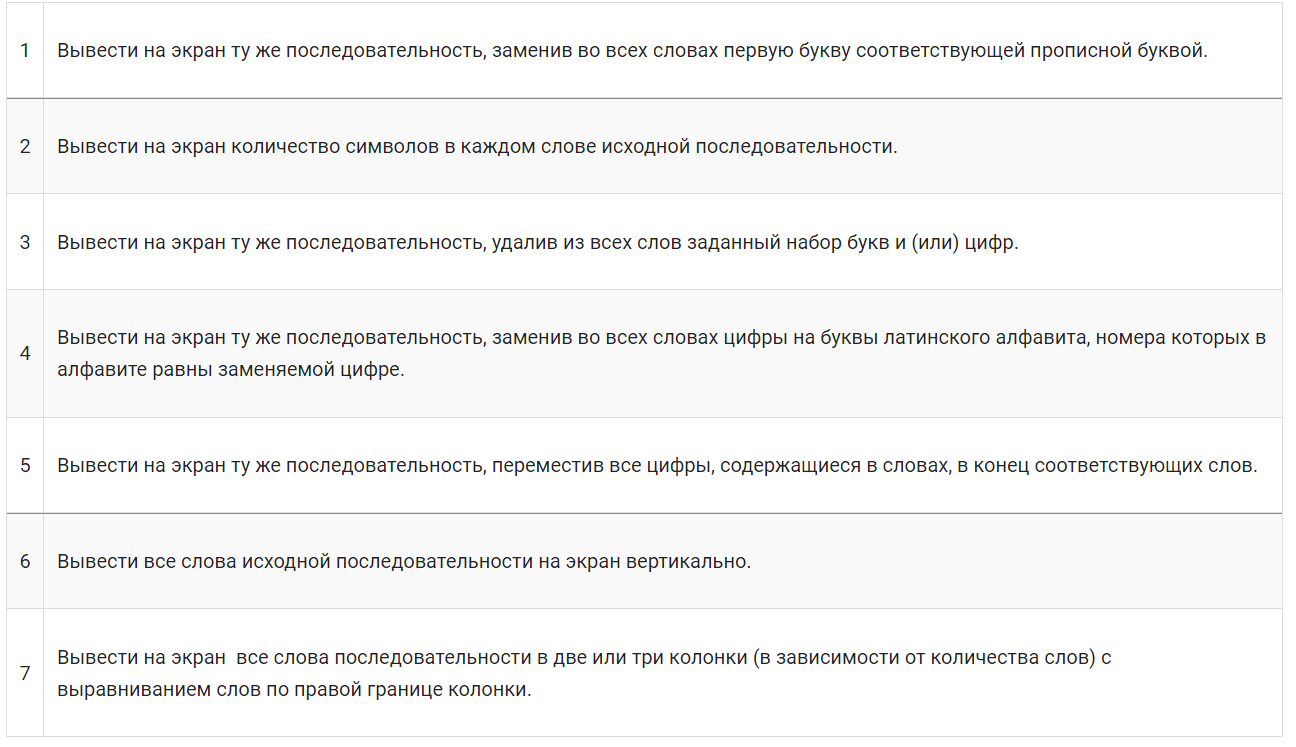
·        удалить лишние знаки препинания (под «лишними» подразумевается несколько подряд идущих знаков (обратите внимание, что «…» - корректное использование знака) в тексте);

·        исправить регистр букв, если это требуется (пример некорректного использования регистра букв: пРиМЕр);

1. Выполнить задание по варианту:



1. Выполнить задание по варианту:

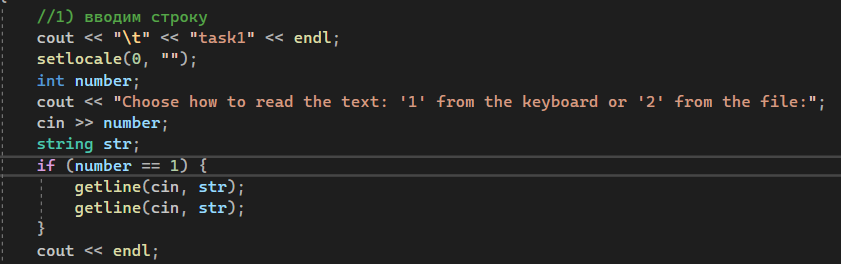


1. Необходимо найти все подстроки, которую введёт пользователь в имеющейся строке. Реализуйте два алгоритма: первый алгоритма – Линейный поиск, а второй алгоритм согласно вашему номеру в списке. Четные номера должны реализовать алгоритм КМП, а нечетные – Бойера-Мура. (\*)

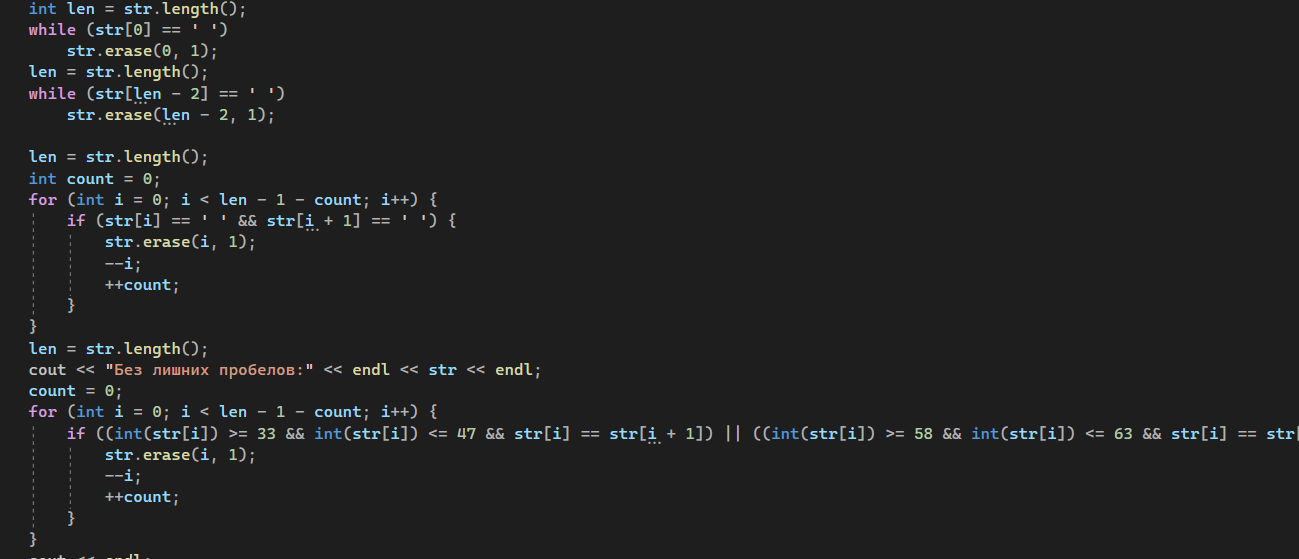
**Выполнение работы.**

Алгоритм выполнения программы:

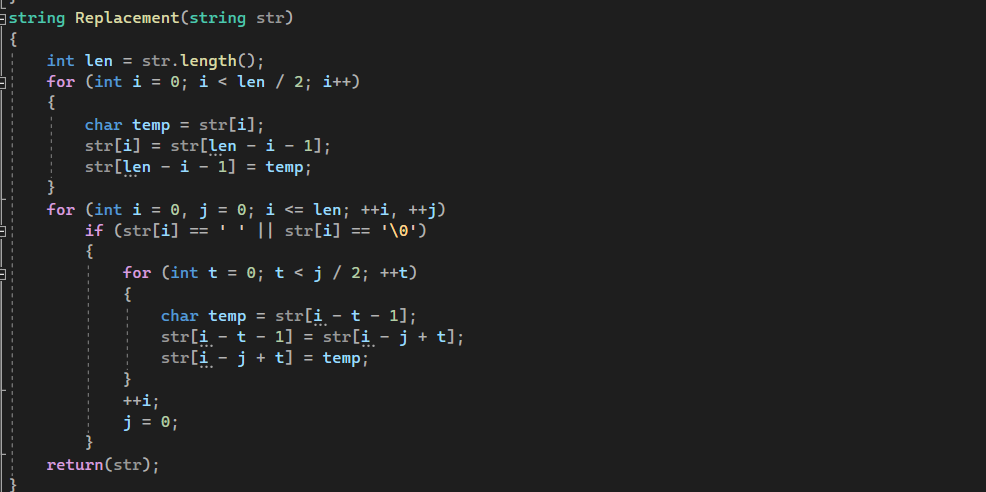
1. Создаем строку



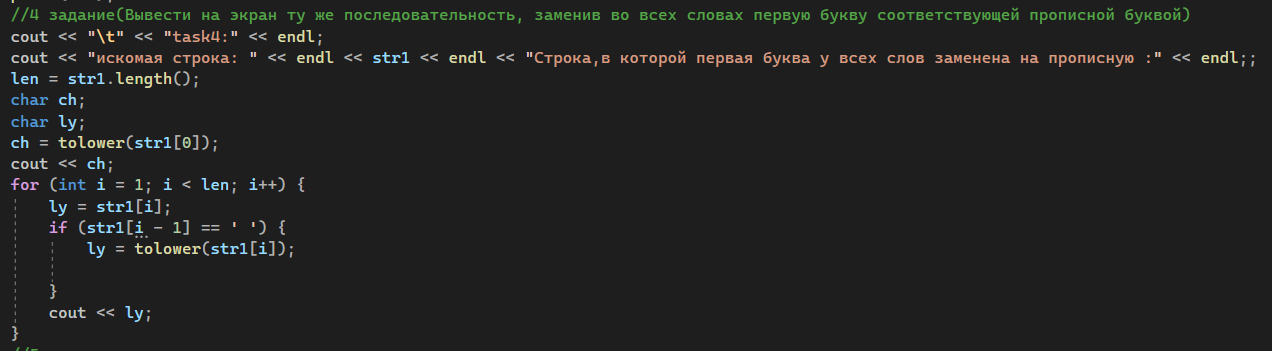
1. Редактируем строку



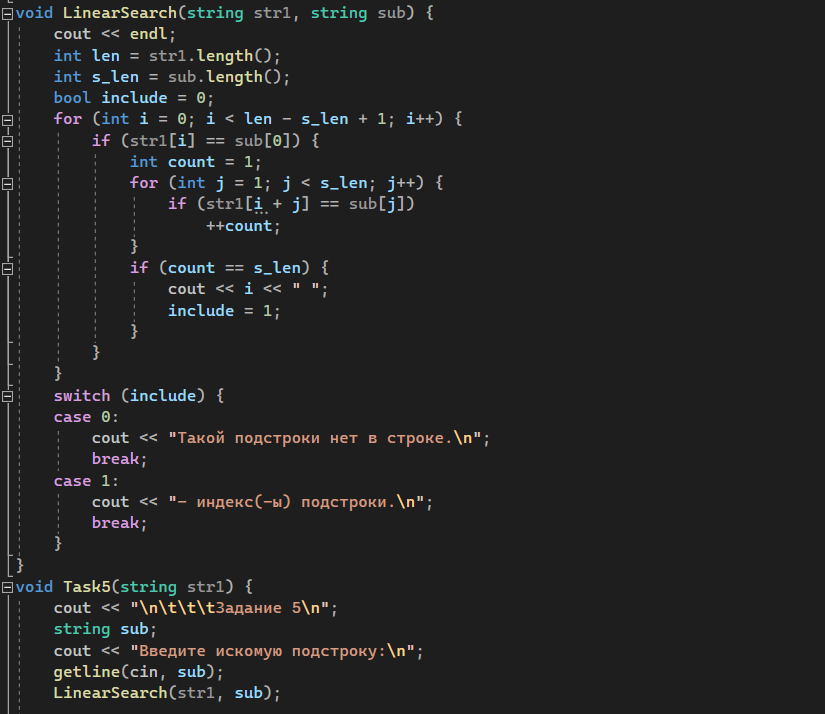
1. Выводим на экран слова последовательности в обратном порядке с помощью функции.



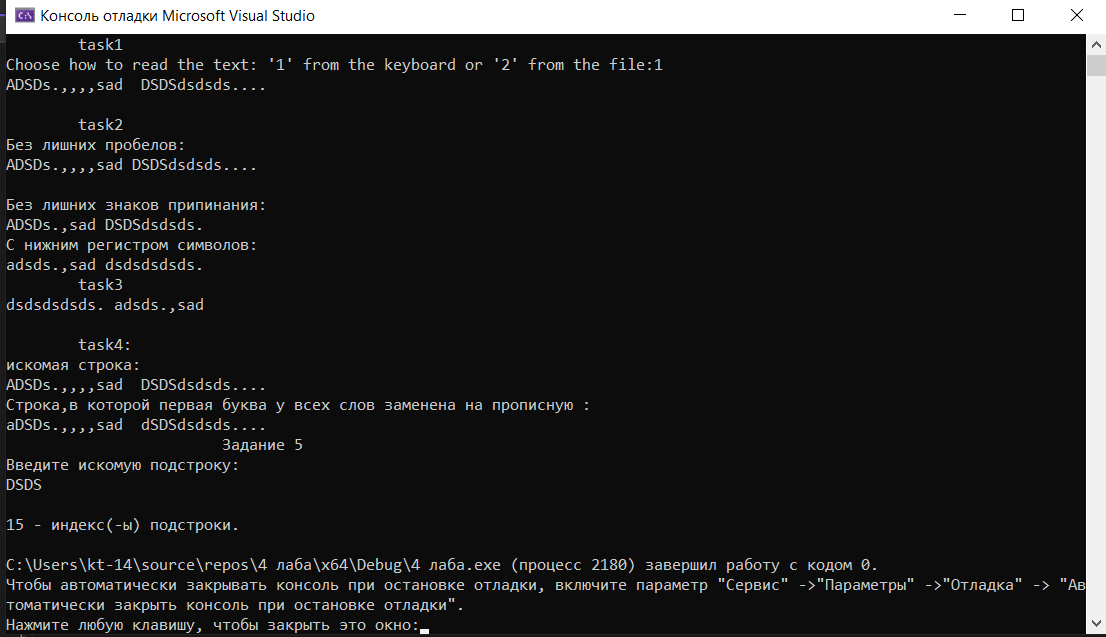
1. Выводим  на экран ту же последовательность, заменив во всех словах первую букву соответствующей прописной буквой.



1. Ищем подстроку в строке.



Пример выполнения работы:



**Выводы.**

Я ознакомился с понятием строк. Изучил теорию работы с ними.