

Лабораторная работа №6

В заданиях рассматривать строки как одномерный или двумерный массив символов char. Обработать неверный пользовательский ввод. Для тестирования программы пользуйтесь базовыми принципами тестирования ПО используя Google Tests, создав по 10 тестов, 2 из которых обязаны покрывать краевые случаи. Лабораторную оформить одним решением с использованием методов. Каждый метод обязан принимать хотя бы один аргумент, там, где это может быть возможно, аргумент обязан являться динамическим массивом. Подготовить к каждому заданию блок-схему. Все задания должны запускаться через Stake с указанием address и thread санитайзеров в скрипте Stake. В работе используйте Linter для языка программирования C/C++.

Задание 1.

При выполнении задания предусмотреть возможность ввода текста с клавиатуры. При работе с текстом использовать строки длиной не более 80-ти символов. Исходный текст и полученный результат вывести на экран. Считывание элементов производится через метод getchar().

1. В строке, состоящей из групп нулей и единиц, разделенных пробелами, найти количество групп с пятью цифрами.
2. В строке, состоящей из групп нулей и единиц, найти и вывести на экран самую короткую группу.
3. В строке, состоящей из групп нулей и единиц, подсчитать количество символов в самой длинной группе.
4. В строке, состоящей из групп нулей и единиц, найти и вывести на экран группы с четным количеством символов.
5. В строке, состоящей из групп нулей и единиц, подсчитать количество единиц в группах с нечетным количеством символов.
6. Из строки, состоящей из букв, цифр, запятых, точек, знаков + (плюс) и – (минус), выделить подстроку, которая соответствует записи целого числа.
7. Из строки, состоящей из букв, цифр, запятых, точек, знаков + (плюс) и – (минус), выделить подстроку, задающую вещественное число с фиксированной точкой.
8. Из строки, состоящей из букв, цифр, запятых, точек, знаков + (плюс) и – (минус), выделить подстроку, задающую вещественное число с плавающей точкой.
9. Дана строка символов, состоящая из цифр, разделенных пробелами. Вывести на экран числа этой строки в порядке возрастания их значений.
10. Дана строка символов, состоящая из цифр, разделенных пробелами. Вывести четные числа этой строки.

11. Дана строка, состоящая из слов, разделенных пробелами. Вывести на экран порядковый номер слова, содержащего k -ю позицию, если в k -й позиции пробел, то вывести номер предыдущего слова. Значение k ввести с клавиатуры.

12. Дана строка, состоящая из слов, разделенных пробелами. Разбить исходную строку на две подстроки, причем первая длиной k символов, если на k -ю позицию попадает слово, то его следует отнести ко второй строке. Значение k вводится с клавиатуры.

13. Дана строка, состоящая из слов, разделенных пробелами. Вывести на экран порядковый номер слова максимальной длины и номер позиции строки, с которой оно начинается.

14. Дана строка, состоящая из слов, разделенных пробелами. Вывести на экран порядковый номер слова минимальной длины и количество символов в этом слове.

15. В строке символов посчитать количество скобок различного вида.

Задание 2.

1. Напечатать квитанцию об оплате за телеграмму, если известна стоимость одного слова.

2. В тексте слова заданной длины заменить указанной подстрокой, длина которой может не совпадать с длиной слова.

3. В тексте каждую букву заменить ее номером в алфавите.

4. В тексте после указанного символа вставить подстроку.

5. После каждого слова текста, оканчивающегося заданной подстрокой, вставить указанный символ.

6. В тексте удалить указанный символ везде, где он встречается.

7. Из текста удалить все символы, не являющиеся буквами, кроме пробелов.

8. Из текста удалить все слова заданной длины, начинающиеся с согласных букв.

9. Найти, каких букв в тексте больше – гласных или согласных.

10. В тексте найти и вывести на экран все слова максимальной и минимальной длины.

11. В тексте найти и напечатать слова, начинающиеся и оканчивающиеся гласной буквой.

12. В тексте найти и напечатать символы, встречающиеся наиболее часто.

13. В тексте нет слов, начинающихся одинаковыми буквами. Напечатать слова текста в таком порядке, чтобы последняя буква каждого слова совпадала с первой буквой последующего слова. Если все слова нельзя напечатать в таком порядке, найти такую цепочку, состоящую из наибольшего количества слов.

14. Найти наибольшее количество предложений текста, в которых есть одинаковые слова.

15. Напечатать без повторения слова текста, у которых первая и последняя буквы совпадают.

Задание 3.

1. Ввести матрицу. Элементы матрицы – строки. Подсчитать сумму кодов символов каждого слова и, если сумма оказалась четной, развернуть зеркально это слово в строке. Полученные матрицы вывести на экран. Стандартных функций работы со строками не использовать.

2. Обработать массив строк (длина каждой строки не более 100 символов). Если в строке встречена последовательность одинаковых символов, заменить их кодом 255, за которым следует код этого символа и количество одинаковых символов. Стандартных функций работы со строками не использовать.

3. Написать программу, которая, анализируя массив, состоящий из строк, выводит на экран количество слов женского, мужского и среднего рода. Стандартных функций работы со строками не использовать.

4. Имеется массив строк. На том же месте, не заводя других массивов, записать слова в обратном порядке, рассматривая все строки, т.е. первое слово первой строки станет последним словом последней строки (если объединить строки, будет гораздо проще). Стандартных функций работы со строками не использовать.

5. В массиве строк найти среднее значение длины строки. Строки, длина которых больше среднего, – обрезать, меньше – добавить пробелы. Полученный массив вывести на экран. Стандартных функций работы со строками не использовать.

6. Ввести строку символов, содержащую два или более слов, разделенных пробелами. Написать функцию, меняющую местами все четные и нечетные слова в строке, предполагая, что за один раз можно менять местами не более двух символов. Результат вывести на экран. Стандартных функций работы со строками не использовать.

7. Написать функции (копирование, сравнение, объединение) для работы со строками, используя указатели. Стандартных функций работы со строками не использовать.

8. Используя указатели, написать функции перевода целых чисел в строку и строки в число; функции перевода вещественных чисел в строку, перевода строки в вещественное число. Стандартных функций работы со строками не использовать.

9. Ввести массив строк. В функции для каждой строки проверить, является она симметричной или нет. (Симметричной считается строка, которая одинаково читается слева направо и справа налево). Вывести на экран саму строку и результат её обработки. Стандартных функций работы со строками не использовать.

10. Ввести массив строк. В функции для каждой строки проверить, является она палиндромом (симметричной с точностью до пробелов) или нет. Например, палиндромами являются цепочки: «АРГЕНТИНА МАНИТ НЕ ГРА» и «А РОЗА УПАЛА НА ЛАПУ АЗОРА». Результат вывести на экран. Стандартных функций работы со строками не использовать.

11. Ввести строку и вычислить сумму входящих в неё цифр, причем знак очередного слагаемого должен быть противоположным знаку предыдущего слагаемого. Например: для строки «asdd1vnb24vnf63vbn,-5h-2kk» сумма $S=1-2+4-6+3-5+2=-3$. Результат вывести на экран. Стандартных функций работы со строками не использовать.

12. Ввести строку и определить наибольшее записанное в этой строке целую часть числа (без учета знака числа). Например, для строки «sdfvgsd1.9fdmjgvb15.25dnj05» наибольшее целое число 15. Результат вывести на экран. Стандартных функций работы со строками не использовать.

13. Ввести строку и определить все входящие в неё символы. Например: строка «abscbbbabba» состоит из символов «a», «b» и «c». Результат вывести на экран. Стандартных функций работы со строками не использовать.

14. Ввести строки символов. Среди литер этого текста особую роль играет знак #, появление которого означает отмену предыдущей литеры текста; k знаков # отменяют k предыдущих литер (если такие есть). Преобразовать строку с учетом роли знака #. Например, строка «VR##Y#HELO#LO» должна быть напечатана в виде: «HELLO». Результирующую строку вывести на экран.

Стандартных функций работы со строками не использовать.

15. Ввести две строки. В функции определить, являются ли они анаграммами (т.е. одна строка получена из другой перестановкой букв). Например: строки БУК и КУБ или СОЛЬ и ЛОСЬ являются анаграммами. Стандартных функций работы со строками не использовать.

Темы для подготовки к теоретическим вопросам: Библиотека на основе функций, библиотека на основе методов класса, аргументы командной строки, ASCII, UTF, строковые алгоритмы, отличия компиляторов gcc/Clang/MS Visio/g++.

При желании получить высокие результаты на защите убедитесь, что в тестирующей системе **stepik** выполнены все задания до пункта 9.2 включительно.