

Самостоятельная работа 5.

Тема. Строки

Цель:

- получение знаний по выполнению операций над строкой, завершающейся нулем;
- получение знаний по организации хранения строковых данных класса `string`;
- получение навыков использования средств языка Си и C++ для реализации алгоритмов обработки текстовых данных.

1. Требования к выполнению практической работы

Требуется выполнить два задания по обработке текста. Варианты задач для обоих заданий представлено в табл. 10.

В первом задании для представления текста использовать нуль терминальную строку языка C++. Для управления строкой использовать функции файла заголовка (библиотека функций управления нуль терминальной строкой) `string.h`.

Во втором задании использовать для представления текста строку класса `string`. Для управления строкой использовать методы класса `string`.

1.1. Задание 1

Разработать программу, согласно задаче варианта, используя для представления текста нуль терминальную строку (строку, завершающуюся нулем) и средства языка C для выполнения операций над этой строкой.

При выполнении операций модификации строки (вставить или удалить подстроку) использовать указатели и функции файла `string.h`.

1.1.1. Требования к выполнению задания 1

1. Изучить функции ввода и вывода текста в строку, завершающуюся нулем.
2. Ознакомиться с функциями файла `string.h`, управления строкой, завершающейся нулем.
3. Требования к разрабатываемой программе.
 - 1) при реализации задачи варианта разработать следующие функции управления строкой:
 - ввод и вывод выполнять функциями для строк, завершающихся нулем;

- выделение слов (часть текста между разделителями) выполнять функцией `strtok` (с коррекцией компилятора);
 - выделение подстроки в тексте выполнять функциями копирования;
 - операции удаления подстроки или вставки подстроки в строку выполнить применяя указатели и функцию конкатенации;
 - массив из слов (подстрок) формировать на основе свободного массива;
- 2) выполнить декомпозицию задачи варианта (т.е. определить подзадачи, которые требуется разработать) определить, какие из них могут быть решены только средствами функций файла `string.h`, а какие должны быть реализованы разработанным алгоритмом с применением функций файла `string.h`;
 - 3) разработать тесты для тестирования программы в целом;
 - 4) разработать программу, управляющую процессом посредством текстового меню;
 - 5) выполнить тестирование программы на подготовленных тестах.

Примечание. Под термином *слово*, в задачах варианта, будем понимать последовательность символов, заключенную между символами разделителями.

1.2. Задание 2

Разработайте программу согласно задаче варианта, используя для представления обрабатываемого в программе текста строку `string`, стандартной библиотеки шаблонов и возможности класса для выполнения операций со строкой.

1.2.1. Требования к выполнению задания 2

1. Изучить функции ввода и вывода текста в строку класса `string`.
2. Изучить набор перегруженных операторов (знаки операций, которые можно применять к объектам класса `string`).
3. Изучить методы класса `string`, по модификации строки, извлечению подстрок строки и др.
4. Требования к разрабатываемой программе.
 - 1) ввод и вывод выполнять функциями строк класса `string`;
 - 2) реализовать задачи, полученные в результате декомпозиции решаемой задачи в задании 1 данной практической работы, при этом, определить, какие из них могут быть решены только средствами класса `string`, а какие должны быть разработаны вами с применением методов и операторов класса `string`; разработать алгоритмы подзадач, требующих разработки, реализовать с применением методов и операторов класса `string`;

- 3) разработать программу, управляющую процессом посредством текстового меню;
- 4) выполнить тестирование программы на подготовленных в задании 1 тестах.

1.3. Варианты задач

Таблица 10. Варианты задач к заданиям 1 и 2 практической работы 4

№	Условие задачи
1.	Дано предложение, составленное из слов и групп цифр (считать тоже словами), определяющих целые числа. Слова разделены запятой, одним или несколькими пробелами. Удалить из предложения слова, которые встретились там более одного раза и сформировать массив из чисел, встретившихся в тексте.
2.	Даны две строки, состоящие только из цифр. Считая, что в этих строках находятся очень длинные целые числа, сформировать третью строку - сумму этих чисел.
3.	Дано предложение, в котором содержатся группы цифр (каждая не более 6 цифр) - это целое десятичное число представленное в строковом формате. Группы отделяются друг от друга пробелами. Найти эти группы, преобразовать в числовой формат своим, а не системным алгоритмом.
4.	Дано предложение, состоящее из слов, разделенных запятой или пробелами среди которых есть группы цифр, определяющих целые числа из диапазона 0..65535. Удалить из текста все числа, принадлежащие диапазону [-100..100] и на их место поставить символ *.
5.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. Среди слов встречаются слова, представляющие код некоторого десятичного числа в 8-ой системе счисления (признака восьмеричного кода в тексте: число начинается с символа \$: само число составлено из цифр '0'..'7'). Сформировать массив из чисел восьмеричного кода предварительно переведя их в десятичную систему счисления.
6.	Дано предложение, состоящее из слов, разделенных запятой или пробелами, среди которых есть группы цифр, определяющих целые числа из диапазона int. Удалить из предложения те целые числа, в десятичной записи которых есть цифры 5,6,7, а остальные числа увеличить на 2.
7.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. Вывести слова предложения предварительно преобразовав их следующим образом: <ul style="list-style-type: none"> - перенести последнюю букву в начало слова; - удалить из слова повторные вхождения каждой буквы.
8.	Дано предложение, состоящее из слов, разделенных запятой или пробелами, среди которых есть слова, составленные только из цифр, определяющие целые числа из диапазона 0 .. 65535. Удалить из предложения все слова-числа, состоящие из одинаковых цифр. Сформировать массив из чисел предложения, преобразовав их в обратные (например: исходное число 123, а в массив записать 321).

9.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. Распечатать те слова предложения, в которых буквы упорядочены по алфавиту, и удалить те слова, в которых каждая буква входит в слово не менее двух раз.
10.	Дан произвольный текст, состоящий из слов. Отредактировать его, оставив между словами по одному пробелу, а между предложениями по два. Предложения завершаются символами точка, вопросительный или восклицательный знак.
11.	Дано предложение и управляющий символ, значением которого может быть один из символов Y, N. Вывести этот текст без входящих в него цифр, если значение символа управления = Y, а если этот символ = N, то перенести все цифры текста в конец предложения так, чтобы первая встреченная в исходном тексте цифра была последней цифрой, а последняя первой и был сохранен порядок следования остальных цифр.
12.	Дано предложение, состоящее из слов, разделенных запятой или пробелами, среди которых есть группы цифр, определяющих целые числа из диапазона [0..19]. Отредактировать введенную строку, заменив каждое число, на последовательность, заключенную в круглые скобки, символа +, (если число четное) или на – (если число нечетное), длина которой равна найденному числу.
13.	Дано предложение, состоящее из символов. Вывести это предложение, удалив из него все символы, которые находятся между символами '(' ')'. Сами скобки не удалять, а вместо удаленного текста вставить число, соответствующее количеству удаленных символов. Если хотя бы одной скобки нет, то сообщить об этом.
14.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. Найти самое длинное слово палиндром. Заменить слово в предложении числом, определяющим его номер среди палиндромов предложения.
15.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. Составить частотный словарь слов, указав, сколько раз данное слово встретилось в предложении. При этом изменить предложение: - удалив повторные вхождения слов и вставив перед первым вхождением слова подстроку вида:(Число), где - Число – определяет количество таких слов в предложении.
16.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. Удалить из предложения все слова, начинающиеся с гласных букв русского алфавита, а слова, начинающиеся с согласных букв записать прописными буквами. Между словами, заканчивающимися и начинающимися одной буквой, вставить подстроку -*.-.
17.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. Среди слов этого предложения найти слова, составленные из тех же букв что и первое слово. Найденные слова переставлять в начало предложения.

Продолжение табл. 10

№	Условие задачи
18.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. В данном предложении есть слова, представляющие запись вещественного числа в формате с плавающей точкой (т.е. +/-xxxx.xxxxE+/-pppp). Создать массив вещественных чисел. В тексте эти слова заменить словами, представляющими это

	число в формате с фиксированной точкой и с заданным (для всех чисел) количеством знаков после точки.
19.	Дано предложение, состоящее из слов, разделенных запятой или пробелами. Слова, которые больше последнего слова, заменить на их перевертыши, а слова, которые меньше последнего, занести в массив слов и удалить из предложения.
20.	Дана последовательность чисел (целых и вещественных). Преобразовать каждое число в строковое представление. Объединить полученные строки в текст разделяя их символом пробел. Объединение строки производить пока длина строки не превысит 255 символов.
21.	Дано предложение, состоящее из слов, разделенных пробелами. Среди слов могут быть числа шестнадцатеричного кода, которые начинаются с символа \$. Сформировать массив из десятичных значений шестнадцатеричных чисел, а сами шестнадцатеричные числа удалить из предложения.
22.	Даны два предложения. Вывести слова, общие для этих двух предложений. Если таких слов нет, то вывести сообщение об этом.
23.	Дан текст программы на языке C++. Вывести все ключевые слова, встретившиеся в тексте программы и указать их количество. Использовать словарь ключевых слов языка C++ (некоторый набор).
24.	Дано предложение, слова в котором разделены пробелами и запятыми. Распечатать те пары слов, расстояние между которыми наименьшее. Расстояние – это количество позиций, в которых слова различаются. Например, МАМА и ПАПА, МЫШКА и КОШКА расстояние этих пар равно двум.
25.	Дано предложение, слова в котором разделены любыми допустимыми знаками препинания. Сформировать массив из слов, в которых заданная подстрока размещается с первой позиции.
26.	Дан текст, слова в котором разделены любыми допустимыми знаками препинания. Сформировать массив из слов, в которых заданная подстрока размещается в конце слова.
27.	Дан текст, который содержит много подстрок (серий) состоящих из одинаковых символов, но в тексте могут быть последовательности в которых нет повторяющихся символов. Выполнить сжатие текста по алгоритму: серии заменить подстрокой вида (к) символ, где к – количество символов в серии, последовательности символов, не являющиеся серией заменить на строку вида (-к) последовательность символов. Например, строка до сжатия AAAABCDEFOOOOOOO строка после сжатия (4)A(-5)BCDEF(7)O. Определите коэффициент сжатия для большого по объему текста. Реализуйте алгоритм восстановления текста.

№	Условие задачи
28.	Дан текст, содержащий слова, которые разделены пробелами. Сформировать массив слов. Найти пары слов (анаграммы)- при прочтении каждого из которых в обратном порядке образуется другое слово этой пары, например, ТОК, КОТ; ПОЛК, КЛОП; БАР, РАБ. Сформировать двумерный массив этих пар.
29.	Дан текст, содержащий слова, которые разделены пробелами. Сформировать массив слов. Найти самое длинное слово, в котором все буквы различны. Например, лейкопластырь, неряшливость, четырёхдюймовка.
30.	Дан текст, содержащий слова, которые разделены пробелами. Найти слова в этом предложении, которые состоят из тех же букв, что и другое слово предложения. Например, ток, кот; лес, сел; гора, рога.
31.	Расстояние между двумя словами равной длины – количество букв, в которых различаются эти слова. В заданном предложении найти пару наиболее далеко удаленных слов заданной длины.

2. Структура отчета

Титульный лист.

Оглавление.

1. Условие задачи и варианта.
2. Концептуальное описание алгоритм решения задачи.
3. Тесты для отладки задачи варианта.
4. Реализация задания 1.
 - 4.1. Описать применяемые в реализации функции над строкой из библиотеки `string.h`.
 - 4.2. Код реализации задания 1.
 - 4.3. Время выполнения функции, реализующей задачу варианта.
 - 4.4. Скрины результатов тестирования.
5. Реализация задания 2.
 - 5.1. Описать применяемые в реализации функции над строкой из библиотеки `<string>` и методы доступа к данным строки.
 - 5.2. Код реализации задания 2.
 - 5.3. Время выполнения функции, реализующей задачу варианта.
 - 5.4. Скрины результатов тестирования.
6. Выводы по результатам выполненной работы.

3. Контрольные вопросы

1. Реализован ли в языке C++ строковый тип как стандартный?

2. Как в программе на C++ определить нуль терминальную строку?

3. Сколько байтов памяти будет выделено для строки s?

```
char s[]="12345";
```

4. Почему не отобразился введенный текст?

```
int main() {  
    char s[10];  
    for(int i=0;i<9;i++)  
        cin>>s[i];  
    cout<<s;    }
```

А что надо изменить в коде, чтобы он отобразился?

5. Как ввести текст, содержащий пробел между словами?

6. Объясните ошибку программиста, сформировавшего следующий код:

```
char s1[10]= "12345", s2[10];  
s2=s1;
```

Исправьте код, чтобы он был корректным.

7. Что будет выведено на экран после выполнения нижеследующих операторов?

```
char s[10]= "12345678";  
cout<<(s+2)<<endl;
```

8. Напишите функцию, которая копирует часть строки, начиная с позиции i (i>0) до j (j<длины строки). Используя только указатели.

9. Как вы считаете, почему файл string.h не содержит функций вставки подстроки в строку и удаления подстроки?

10. Опишите алгоритм выполнения действия функцией strtok. Продемонстрируйте на примере.

11. Приведите объявление переменной для хранения массива строк.

12. Выполните представленный код. Результатом является вывод значения переменной s1. Объясните получение результата.

```
char s1[]="123 456,789",*s2;  
s2=strtok(s1," ,");  
while(s2) {s2=strtok(NULL," ,");}  
cout<<s1;
```

13. Приведите эффективный по количеству операторов код функции, которая определит количество вхождений заданной подстроки в исходную строку.

14. Приведите примеры применения функций strtod, atoi.

15. Расскажите о формате представления строк string.

16. Какую информацию о строке `string` хранит свойство `capacity`, а какую `max_size`?
17. Каков максимально возможный размер обрабатываемой строки в программе C++?
18. Является ли `string` строка динамической?
19. Как создать динамическую строку, завершающуюся нулем?
20. Какую информацию хранит свойство `npos` строки `string`?
21. Приведите пример использования функции *reserve* строки `string`.