

## Лабораторная работа №6

### Обработка строк

**Цель работы:** получение практических навыков при работе со строками.

#### Основные функции стандартной библиотеки `string.h`

Основные функции стандартной библиотеки `string.h` приведены в таблице.

Функция	Описание
<code>char *strcat(char *s1, char *s2)</code>	присоединяет <code>s2</code> к <code>s1</code> , возвращает <code>s1</code>
<code>char *strncat(char *s1, char *s2, int n)</code>	присоединяет не более <code>n</code> символов <code>s2</code> к <code>s1</code> , завершает строку символом <code>'\0'</code> , возвращает <code>s1</code>
<code>char *strcpy(char *s1, char *s2)</code>	копирует строку <code>s2</code> в строку <code>s1</code> , включая <code>'\0'</code> , возвращает <code>s1</code>
<code>char *strncpy(char *s1, char *s2, int n)</code>	копирует не более <code>n</code> символов строки <code>s2</code> в строку <code>s1</code> , возвращает <code>s1</code> ;
<code>int strcmp(char *s1, char *s2)</code>	сравнивает <code>s1</code> и <code>s2</code> , возвращает значение 0, если строки эквивалентны
<code>int strncmp(char *s1, char *s2, int n)</code>	сравнивает не более <code>n</code> символов строк <code>s1</code> и <code>s2</code> , возвращает значение 0, если начальные <code>n</code> символов строк эквивалентны
<code>int strlen(char *s)</code>	возвращает количество символов в строке <code>s</code>
<code>char *strset(char *s, char c)</code>	заполняет строку <code>s</code> символами, код которых равен значению <code>c</code> , возвращает указатель на строку <code>s</code>
<code>char *strnset(char *s, char c, int n)</code>	заменяет первые <code>n</code> символов строки <code>s</code> символами, код которых равен <code>c</code> , возвращает указатель на строку <code>s</code>

## Пример программы:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main() {
    char m1[80] = "First string";
    char m2[80] = "Second string";
    char m3[80];
    printf("The string m1: ");
    puts(m1);
    printf("The string m2: ");
    puts(m2);
    strncpy(m3, m1, 6); // не добавляет '\0' в конце строки
    puts("The result of strncpy(m3, m1, 6) ");
    puts(m3);
    strcpy(m3, m1);
    puts("The result of strcpy(m3, m1) ");
    puts(m3);
    puts("The result of strcmp(m3, m1) ");
    printf("%d", strcmp(m3, m1));
    strncat(m3, m2, 5);
    puts("The result of strncat(m3, m2, 5)");
    puts(m3);
    strcat(m3, m2);
    puts("The result of strcat(m3, m2)");
    puts(m3);
    puts("Symbols in the string m1 is strlen(m1) : ");
    printf("%d\n", (int)strlen(m1));
    _strnset(m3, 'f', 7);
    puts("The result of strnset(m3, 'f', 7)");
    puts(m3);
    _strset(m3, 'k');
    puts("The result of strnset(m3, 'k')");
    puts(m3);
    getchar();
    return 0;
}
```

```
C:\УМК\ОАиП ПИН\УМК Си\ЛК\ЛК 10 примеры\2.exe
The string m1: First string
The string m2: Second string
The result of strncpy(m3, m1, 6)
First
The result of strcpy(m3, m1)
First string
The result of strcmp(m3, m1)
0The result of strncat(m3, m2, 5)
First stringSecon
The result of strcat(m3, m2)
First stringSeconSecond string
Symbols in the string m1 is strlen(m1) :
12
The result of strnset(m3, 'f', 7)
fffffffftrngSeconSecond string
The result of strnset(m3, 'k')
kkkkkkkkkkkkkkkkkkkkkkkkkkkkkkkk
```

## Задания на лабораторную работу 6

### Задание 1. Простейшие задачи

1. Составить программу, которая:
  - а) запрашивает имя человека и повторяет его на экране;
  - б) запрашивает имя человека и повторяет его на экране с приветствием.
2. Составить программу, которая запрашивает название футбольной команды и повторяет его на экране со словами "— это чемпион!".
3. Составить программу, которая запрашивает отдельно имя и отдельно фамилию, а затем выводит их как одну символьную строку.
4. Составить программу, которая запрашивает название государства и его столицы, а затем выводит сообщение: "Столица государства ... — город ..." (на месте многоточий должны быть выведены соответствующие значения).
5. Составить программу, которая запрашивает название романа и фамилию его автора, а затем выводит сообщение: "Писатель ... — автор романа ..." (на месте многоточий должны быть выведены соответствующие значения).
6. Даны названия двух стран. Присвоить эти названия переменным величинам s1 и s2, после чего название s2 присвоить величине t1, название s1 — величине t2.

7. Дано название футбольного клуба. Определить количество символов в нем.
8. Дано название города. Определить, четно или нет количество символов в нем.
9. Даны две фамилии. Определить, какая из них длиннее.
10. Даны названия трех городов. Вывести на экран самое длинное и самое короткое название.
11. Даны названия двух стран. Присвоить эти названия переменным величинам  $s1$  и  $s2$ , после чего обменять значения величин  $s1$  и  $s2$ .
12. Составить программу обмена значениями трех переменных величин  $a$ ,  $b$ ,  $c$  строкового типа по следующей схеме:  $b$  присвоить значение  $c$ ,  $a$  присвоить значение  $b$ ,  $c$  присвоить значение  $a$ ;
13. Составить программу обмена значениями трех переменных величин  $a$ ,  $b$ ,  $c$  строкового типа по следующей схеме:  $b$  присвоить значение  $a$ ,  $c$  присвоить значение  $b$ ,  $a$  присвоить значение  $c$ .

## **Задание 2. Работа с символами строки**

1. Из слова *яблоко* путем "вырезок" и "склеек" его букв получить слова *блок* и *око*.
2. Из слова *информатика* путем "вырезок" и "склеек" его букв получить слова *форма* и *тик*.
3. Из слова *вертикаль* путем "вырезок" и "склеек" его букв получить слова *тир* и *ветка*.
4. Из слова *программа* путем "вырезок" и "склеек" его букв получить слова *ром* и *рампа*.
5. Из слова *трос* путем "вырезок" и "склеек" его букв получить слова *сорт*, *рост* и *торс*.
6. Из слова *клоун* путем "вырезок" и "склеек" его букв получить слова *уклон*, *кулон* и *колун*.

7. Из слова *апельсин* путем "вырезок" и "склеек" его букв получить слово *спаниель*.

8. Из слова *вирус* путем замены его букв получить слово *фокус*.

9. Из слова *курсор* путем замены его букв получить слово *танцор*.

10. Из слова *пробел* путем замены его букв получить слово *продел*.

11. Из слова *строка* путем замены его букв получить слово *строфа*.

12. Из слова *муха* путем замены его букв получить слово *слон*.

13. Из слова *тетрадь* путем замены его букв получить слово *дневник*.

### **Задание 3. Обработка строк с использованием оператора цикла с параметром**

1. Дано предложение. Напечатать все его буквы *и*.

2. Дано предложение. Составить программу, которая печатает "столбиком" все вхождения в предложение некоторого символа.

3. Дано предложение. Вывести "столбиком" его третий, шестой и т. д. символы.

4. Дано предложение. Вывести все буквы *м* и *н* в нем.

5. Дано предложение. Составить программу, которая выводит все вхождения в предложение двух заданных символов.

6. Дано предложение. Вывести все имеющиеся в нем буквосочетания *nn*.

7. Дано предложение. Вывести "столбиком" все его буквы *и*, стоящие на четных местах.

8. Дано предложение. Вывести "столбиком" его первый, второй, пятый, шестой,

девятый, десятый и т. д. символы.

9. Дано предложение. Определить число букв *о* в нем.

10. Дано предложение. Определить число пробелов в нем.

11. Дано предложение. Определить число вхождений в него некоторого символа.

12. Дано предложение. Определить долю (в %) букв *а* в нем.

13. Дан текст. Сколько раз в нем встречается символ "+" и сколько раз символ "\*"?

**Задание 4. Обработка строк с использованием операторов цикла с условием**

1. Дано предложение. Напечатать все его символы, предшествующие первой запятой. Рассмотреть два случая:

- 1) известно, что в предложении запятые имеются;
- 2) в предложении запятых может не быть.

2. Дано предложение, в котором имеется несколько букв *e*. Найти:

- а) порядковый номер первой из них;
- б) порядковый номер последней из них.

3. Дано предложение. Определить, есть ли буква *a* в нем. В случае положительного ответа найти также порядковый номер первой из них.

4. Дано слово. Проверить, является ли оно "перевертышем" (*перевертышем* называется слово, читаемое одинаково как с начала, так и с конца).

5. Дан текст. Определить количество букв *и* в первом предложении. Рассмотреть два случая:

- 1) известно, что буквы *и* в этом предложении есть;
- 2) букв *и* в тексте может не быть.

6. Дана последовательность символов, в начале которой имеется некоторое количество одинаковых символов. Определить это количество. Рассмотреть два случая:

- 1) известно, что не все символы последовательности одинаковые;
- 2) все символы последовательности могут быть одинаковыми.

7. Даны два слова. Определить, сколько начальных букв первого слова совпадает с начальными буквами второго слова.

Рассмотреть два случая:

- 1) известно, что слова разные;

2) слова могут быть одинаковыми.

8. Дано предложение, в котором нет символа "-". Определить количество букв *o* в первом слове. Учесть, что в начале предложения могут быть пробелы.

9. Дано предложение. Определить количество букв *n*, предшествующих первой запятой предложения. Рассмотреть два случая:

1) известно, что запятые в предложении есть;

2) запятых в предложении может не быть.

10. Дано предложение. Определить порядковые номера первой пары одинаковых соседних символов. Если таких символов нет, то должно быть напечатано соответствующее сообщение.

11. Дано предложение. Определить, есть ли в нем буквосочетания *чу* или *щу*. В случае положительного ответа найти также порядковый номер первой буквы первого из них.

12. Дана последовательность слов. Проверить, правильно ли в ней записаны буквосочетания *жи* и *ши*.

13. Дана последовательность слов. Проверить, правильно ли в ней записаны буквосочетания *ча* и *ща*. Исправить ошибки.

### **Задание 5\*. Задачи повышенной сложности**

1. Даны два слова. Для каждой буквы первого слова (в том числе для повторяющихся в этом слове букв) определить, входит ли она во второе слово.

Например, если заданные слова *информация* и *процессор*, то для букв первого из них ответом должно быть: *нет нет нет да да нет нет да нет нет*.

2. Даны два слова. Для каждой буквы первого слова определить, входит ли она во второе слово. Повторяющиеся буквы первого слова не рассматривать. Например, если заданные слова *процессор* и *информация*, то для букв первого из них ответом должно быть: *нет да да да нет нет*.

3. Даны два слова. Напечатать только те буквы слов, которые есть только в одном из них (в том числе повторяющиеся). Например, если заданные слова *процессор* и *информация*, то ответом должно быть: *п е с с и ф м а я*.

4. Даны два слова. Напечатать только те буквы слов, которые встречаются в обоих словах только один раз. Например, если заданные слова *процессор* и *информация*, то ответом должно быть: *п е ф м а я*.

5. Даны два слова. Определить, можно ли из букв первого из них получить второе. Рассмотреть два варианта:

1) повторяющиеся буквы второго слова могут в первом слове не повторяться;

2) каждая буква второго слова должна входить в первое слово столько же раз, сколько и во второе.

6. Даны три слова. Напечатать только те буквы слов, которые есть лишь в одном из слов. Рассмотреть два варианта:

1) повторяющиеся буквы каждого слова рассматриваются;

2) повторяющиеся буквы каждого слова не рассматриваются.

7. Даны три слова. Напечатать их общие буквы. Повторяющиеся буквы каждого слова не рассматривать.

8. Даны три слова. Напечатать неповторяющиеся в них буквы.

9. Дано предложение. Напечатать его в обратном порядке слов, например, предложение *мама мыла раму* должно быть напечатано в виде *раму мыла мама*.

10. Дана последовательность слов. Напечатать те слова последовательности, которые отличны от первого слова и удовлетворяют следующему свойству:

а) в слове нет повторяющихся букв;

б) слово симметрично.

11. Даны два предложения. Для каждого слова первого предложения (в том числе для повторяющихся в этом предложении слов) определить, входит ли оно во второе предложение.

12. Даны два предложения. Для каждого слова первого предложения определить, входит ли оно во второе предложение. Повторяющиеся слова первого предложения не рассматривать.



13. Даны два предложения. Напечатать слова, которые есть только в одном из них (в том числе повторяющиеся).