

**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное автономное образовательное учреждение высшего  
образования**

**«Национальный исследовательский университет ИТМО»**

**Факультет информационных технологий и программирования**

**Лабораторная работа № 8**

*Работа со строками*

*Вариант 8*

**Выполнил студент группы № М3111**

**Гонтарь Тимур Сергеевич**

**Подпись:**



Санкт-Петербург  
2022

Условие ЛР:

1. Осуществить конкатенацию (сложение) двух строк.
2. Осуществить сравнение двух строк.
3. Осуществить поиск в строке первого вхождения указанного символа.
4. Осуществить поиск в одной строке любого из множества символов, входящих во вторую строку.
5. Определить длину отрезка одной строки, не содержащего символы второй строки

Необходимо выполнить 5 из указанных 13 заданий согласно номеру варианта. Все входные данные (строки, количество символов) вводить с консоли. Результаты выполнения также вывести на консоль.

Решение с комментариями:

Мой вариант №8

Во всей лабораторной я использую функции из библиотеки string.h

В пункте 1 я использую функцию strcat. Функция записывает вторую строку в конец первой, поэтому я создаю новую строку, копирую в нее 1 строку и доливаю в неё вторую. В пункте 2 я использую функцию strcmp, которая возвращает 0 если строки равны и 1 или -1 если неравны.

В пункте 3 я использую функцию strchr для нахождения первого вхождения символа в строку 1 и в строку 2

В пункте 4 я использую функцию strpbrk для нахождения первого вхождения любого символа строки 2 в строку 1.

В пункте 5 я с помощью указателя я прохожусь циклом по строке 1, и считаю суммарную длину отрезка строки 1, который не содержит символы из строки 2.

```
#include "stdio.h"
#include "string.h"
#include "locale.h"

int main() {
    setlocale( Category: LC_ALL, Locale: "Russian");

    char str1[10];
    char str2[10];
    gets( Buffer: str1);
    gets( Buffer: str2);

    char conct[20];
    strcpy( Dest: conct, Source: str1);
    strcat( Dest: conct, Source: str2);
    printf( format: "Конкатенация: %s\n", conct);

    int comp = strcmp(str1, str2);
    printf( format: "Сравнение: %s\n", (comp == 0) ? ("Строки равны") : ("Строки не равны"));
    printf( format: "Сравнение: %d\n", comp);}

    char symbol;
    printf( format: "Введите символ, первое вхождение которого нужно искать:");
    scanf( format: "%c", &symbol);
    char *frstsmbl = strchr( Str: str1, Val: symbol);
    char *frstsmbl2 = strchr( Str: str2, Val: symbol);
    printf( format: "Первое вхождение символа %c в строку 1 равно: %lld\n", symbol, (frstsmbl == NULL) ? (-1) : (frstsmbl - str1 + 1));
    printf( format: "Первое вхождение символа %c в строку 2 равно: %lld\n", symbol, (frstsmbl2 == NULL) ? (-1) : (frstsmbl2 - str2 + 1));

    char *frstsmbl3 = strpbrk( Str: str1, Control: str2);
```

```

char *frstsm3 = strpbrk( Str: str1, Control: str2);
printf( Format: "Первое вхождение любого символа строки 2 в строку 1 равно: %lld\n", (frstsm3 == NULL) ? (-1) : (frstsm3 - str1 + 1));

int ans = 0;
char *p = str1;
char *n = p + strlen( Str: str1);
while (p < n) {
    int cnt = strcspn( Str: p, Control: str2);
    ans += cnt;
    p += cnt + 1;
}
printf( Format: "Длина отрезка 1 строки, не содержащего символы строки 2: %d", ans);

return 0;
}

```

Вывод: В ходе данной лабораторной работы я использовал различные функции библиотеки string.h для выполнения различных операций на строках: конкатенация, сравнение, поиск первого вхождения и т.д.