

КАФЕДРА №

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ  
ПРЕПОДАВАТЕЛЬ

---

должность, уч. степень, звание

---

подпись, дата

---

инициалы, фамилия

## ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ

### **Обработка текстовых данных**

по курсу: ОСНОВЫ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

---

подпись, дата

---

инициалы, фамилия

Санкт-Петербург 2022

**1.Цель работы:** Целью работы является изучение представления строковых данных в языке С и алгоритмов их обработки.

**2.Задачи работы:** Разработать функцию, которая выполняет ту обработку символьной строки, которая определена в Вашем индивидуальном задании. Все утечки памяти должны быть устранены.

Функция находит в строке первый символ, который не входит в другую заданную строку

### 3.Описание функций:

main-основная функция

1)Имя: main

**Назначение:** ввод пользователем первой и второй строки и поиск символа, который которые не вошли во вторую строку из первой.

**Входные данные:** нет

**Выходные данные:** нет

**Побочный эффект:** отсутствует.

**Тестовые данные:**

**Ввод: hello world**

**hello guar**

**Вывод: Первый символ, который мы не нашли: w**

**Прототип: int main()**

**Псевдокод:**

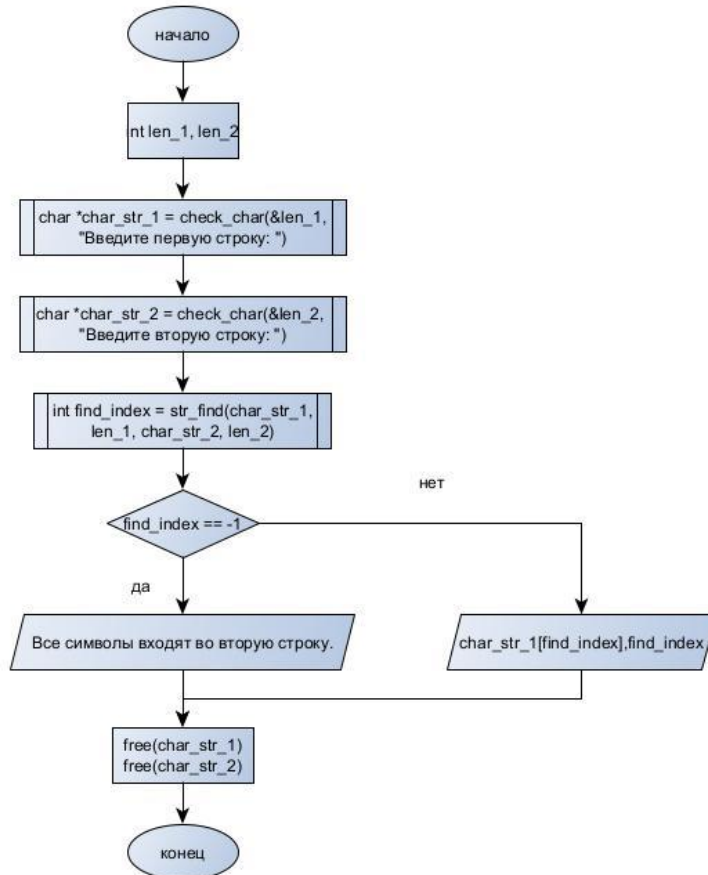
ввод первой строки пользователем

ввод второй строки пользователем

поиск символа, который не входит во вторую строку из первой

если функция такого символа не нашла, то вывод сообщения, что символы входят

**Блок-схема:**



## 2)Имя: check\_char()

**Назначение:** функция для ввода строки и проверки её на заполнение

**Входные данные:** нет

**Выходные данные:** char\_str

**Побочный эффект:** отсутствует.

**Прототип:** char\* check\_char()

**Псевдокод:**

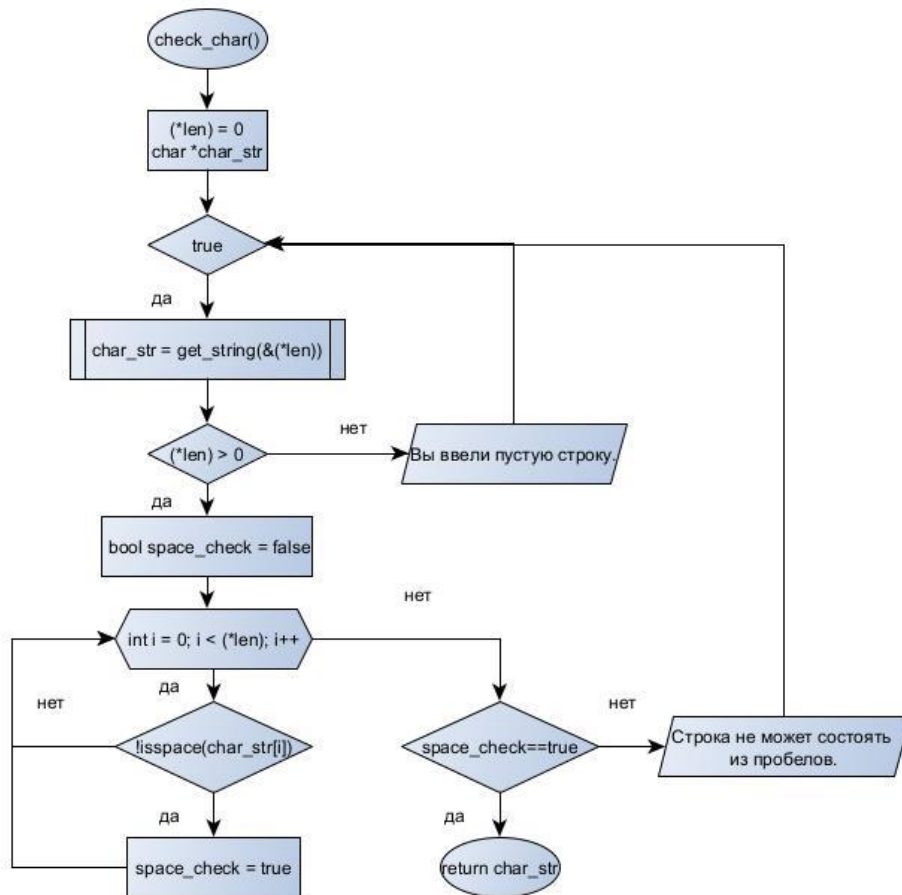
Функция ввода строки

Если символов больше 0, то проверка на пробелы

Если количество пробелов=количество символов, то ввод строки заново

Если символов 0, то ввод строки заново

**Блок-схема:**



## 3)Имя: get\_string()

**Назначение:** функция для ввода строки

**Входные данные:** нет

**Выходные данные:** s

**Побочный эффект:** отсутствует.

**Прототип:** char\* get\_string()

**Псевдокод:**

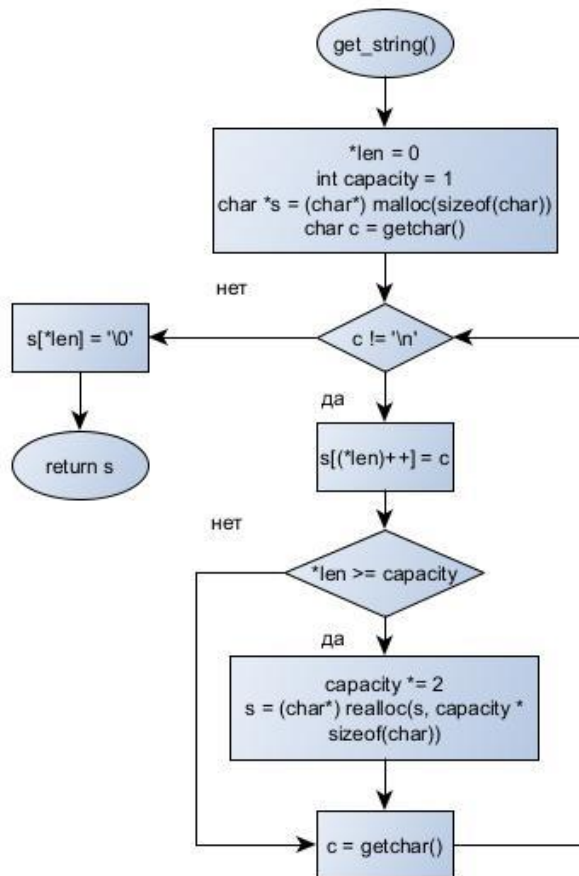
Создание массива

Получение символа

Если этот символ не равен `'\n'`, то запись в массив

Если равен `'\n'`, то конец функции

**Блок-схема:**



#### 4)Имя: str\_find()

**Назначение:** функция нахождения в строке символа, который не входит в другую строку

**Входные данные:** \*str\_1, \*str\_2, len\_1, len\_2

**Выходные данные:** i

**Побочный эффект:** отсутствует.

**Прототип:** int str\_find()

**Псевдокод:**

Создание флага состояния поиска

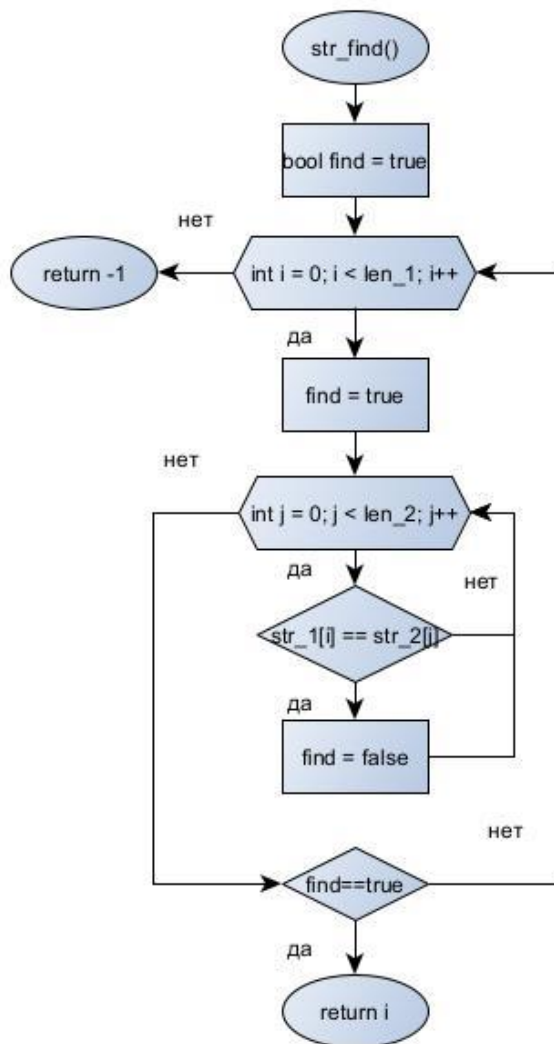
Перебирается первая строка + обновление флага на состояние true

В процессе перебора перебирается вторая строка, если символы совпадают, то флаг=false

Если находится символ, который мы не нашли, то возвращаем его

Если все символы нашли, то возвращаем -1

**Блок-схема:**



## 5.ЛИСТИНГ КОДА:

```

#include <iostream>
using namespace std;

#include "libs/lib.h"
#include <cmath>
#include <time.h>

// проверка ввода
#include "libs/simple_char.h"

#include "more_char.h"
char *get_string(int *len) {
    *len = 0;
    int capacity = 1;
    char *s = (char*) malloc(sizeof(char));
    char c = getchar();
    while (c != '\n') {
        s[(*len)++] = c;
        if (*len >= capacity) {
            capacity *= 2;
            s = (char*) realloc(s, capacity * sizeof(char));
        }
        c = getchar();
    }
    s[*len] = '\0';
}

```

```

        return s;
    }

int str_find(char *str_1, int len_1, char *str_2, int len_2) {
    // флаг состояния поиска
    bool find = true;

    // для 1 строки
    for (int i = 0; i < len_1; i++) {
        find = true;

        // цикл для 2 строки
        for (int j = 0; j < len_2; j++)

            // если нашли одинаковые символы, то идём на следующую итерацию цикла
            if (str_1[i] == str_2[j]) {
                find = false;
                break;
            }

        // если есть символ который мы не нашли, то возвращаем его
        if (find) return i;
    }

    // если строки одинаковые (по набору символов), то возвращаем -1
    return -1;
}

// функция для ввода строки и проверки её
char *check_char(int *len, const char *promt = "") {
    (*len) = 0;
    char *char_str;

    // ввод строки
    while (true) {
        cout << promt;
        char_str = get_string(&(*len));

        if ((*len) > 0) {
            // проверка на пробелы (строка не должна состоять из пробелов)
            bool space_check = false;
            for (int i = 0; i < (*len); i++)
                if (!isspace(char_str[i])) {
                    space_check = true;
                    break;
                }

            if (space_check)
                break;
            else
                cout << "Строка не может состоять из пробелов." << endl;

        } else {
            cout << "Вы ввели пустую строку." << endl;
        }
    }

    return char_str;
}

int main() {
    // смена кодировки
    system("chcp 65001");
}

```

```

// очистка терминала
//clear_scr();

// здесь будет храниться длинна вводимых строк
int len_1, len_2;

// вводим строки
char *char_str_1 = check_char(&len_1, "Введите первую строку: ");
char *char_str_2 = check_char(&len_2, "Введите вторую строку: ");

draw_line(20);

//
int find_index = str_find(char_str_1, len_1, char_str_2, len_2);

if (find_index == -1) {
    cout << "Все символы входят во вторую строку." << endl;
} else {
    cout << "Первый символ который мы не нашли: " << char_str_1[find_index] <<
endl;
    cout << "Его индекс в первой строке: " << find_index << endl;
}

// очищаем память
free(char_str_1);
free(char_str_2);

return 0;
}

```

## 6.Пример выполнения программы:

main

```

C:\Windows\system32\cmd.exe

C:\Users\nikit\Desktop\[ ПРОЕКТЫ ]\basic_programming_GUAP\17 вариант\7>g++ main.cpp -o main.exe

C:\Users\nikit\Desktop\[ ПРОЕКТЫ ]\basic_programming_GUAP\17 вариант\7>main.exe
Active code page: 65001
Введите первую строку: hello world
Введите вторую строку: hello guar
-----
Первый символ который мы не нашли: w
Его индекс в первой строке: 6

C:\Users\nikit\Desktop\[ ПРОЕКТЫ ]\basic_programming_GUAP\17 вариант\7>pause
Press any key to continue . . .

```

## 7.Анализ результатов и выводы:

В ходе этой лабораторной работы мы изучили структуру данных одномерных массивов и перезаписывать их.

Из плюсов можно отметить, что поиск символа происходит с учетом регистра.

Существенных минусов в программе не обнаружено.