МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Факультет автоматики и вычислительной техники

Кафедра автоматики



**ЛАБОРАТОРНАЯ РАБОТА №2**

по дисциплине: «Информатика»

УКАЗАТЕЛИ, ФУНКЦИИ

вариант № 8

Выполнил: Проверил:

Студент гр. АВТ-219 Ядрышников О.Д

Баскаков М.Д.

«\_\_\_» \_\_\_\_\_\_\_\_\_ 2022 г.«\_\_\_» \_\_\_\_\_\_\_\_\_ 2022 г.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка, подпись)

Новосибирск

2022

**Цель работы:**

Освоить правила написания и использования функций в языке СИ. Научиться использовать указатели при обработке массивов данных.

**Исходные данные и задание для варианта:**

4.1. Написать программу сортировки массива строк по вариантам (табл. 2.2.). Ввод данных, сортировку и вывод результатов оформить в виде функций. Входные и выходные параметры функции сортировки указаны в таблице. Входные и выходные параметры функций для ввода-вывода приведены в таблице 2.1.

4.2. Модифицировать программу п.1., применив в функциях передачу параметров и возврат результатов по ссылке (с использованием указателей). Сравнить результаты.

| Вари ант | Задание | Входные  параметры | Выходные  Параметры |
| --- | --- | --- | --- |
| 8 | Расположить строки по убыванию количества цифр | 1. Массив  2. Размерность массива | 1. Количество перестановок  2. Количество цифр |

**Текст программы:**

**Первое задание:**

#include <stdio.h>  
#include <stdlib.h>  
  
int inp\_str(char str[], int maxlen);  
  
void sort(char \*strings[], int count);  
  
int digits\_count(char str[]);  
  
void out\_str(char str[], int length, int number);  
  
void main() {  
 char \*strings[8];  
  
 for (int i = 0; i < 8; i++) {  
 strings[i] = malloc(64);  
 inp\_str(strings[i], 64);  
 }  
  
 sort(strings, 8);  
  
 for (int i = 0; i < 8; i++) {  
 out\_str(strings[i], 0, 0);  
 free(strings[i]);  
 }  
}  
  
int inp\_str(char str[], int maxlen) {  
 int ch, i;  
 for (i = 0; i < maxlen && (ch = getchar()) != '\n'; i++)  
 str[i] = ch;  
 str[i] = 0;  
  
 return 0;  
}  
  
void sort(char \*strings[], int count) {  
 int digitsCount = 0, swaps = 0;  
 for (int i = 0; i < count; i++)  
 digitsCount += digits\_count(strings[i]);  
  
 char swap = 1;  
 while (swap) {  
 swap = 0;  
  
 for (int i = 0; i < count - 1; i++) {  
 if (digits\_count(strings[i]) < digits\_count(strings[i + 1])) {  
 char \*temp = strings[i];  
 strings[i] = strings[i + 1];  
 strings[i + 1] = temp;  
  
 swap = 1;  
 swaps++;  
 }  
 }  
 }  
 printf("Swaps: %d, digits count: %d\n", swaps, digitsCount);  
}  
  
int digits\_count(char str[]) {  
 int count = 0;  
 while (\*str != 0) {  
 if (\*str >= '0' && \*str <= '9')  
 count++;  
 str++;  
 }  
 return count;  
}  
  
void out\_str(char str[], int length, int number) { printf("%s\n", str); }

**Второе задание:**

#include <stdio.h>

#include <stdlib.h>

int inp\_str(char \*str, int maxlen);

void sort(char \*strings[], int count, int \*swaps, int \*digitsCount);

int digits\_count(char \*str);

void out\_str(char \*str, int length, int number);

void main() {

char \*strings[8];

for (int i = 0; i < 8; i++) {

strings[i] = malloc(64);

inp\_str(strings[i], 64);

}

int swaps, digitsCount;

sort(strings, 8, &swaps, &digitsCount);

printf("Swaps: %d, digits count: %d\n", swaps, digitsCount);

for (int i = 0; i < 8; i++) {

out\_str(strings[i], 0, 0);

free(strings[i]);

}

}

int inp\_str(char \*str, int maxlen) {

int ch, i;

for (i = 0; i < maxlen && (ch = getchar()) != '\n'; i++)

str[i] = ch;

str[i] = 0;

return 0;

}

void sort(char \*strings[], int count, int \*swaps, int \*digitsCount) {

\*digitsCount = 0;

for (int i = 0; i < count; i++)

\*digitsCount += digits\_count(strings[i]);

char swap = 1;

while (swap) {

swap = 0;

for (int i = 0; i < count - 1; i++) {

if (digits\_count(strings[i]) < digits\_count(strings[i + 1])) {

char \*temp = strings[i];

strings[i] = strings[i + 1];

strings[i + 1] = temp;

swap = 1;

(\*swaps)++;

}

}

}

}

int digits\_count(char \*str) {

int count = 0;

while (\*str != 0) {

if (\*str >= '0' && \*str <= '9')

count++;

str++;

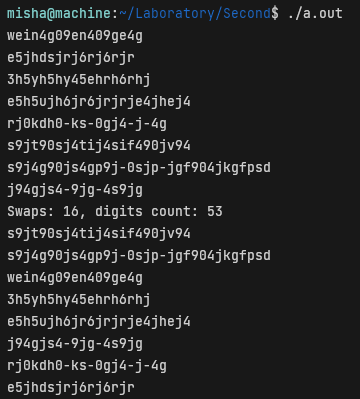
}

return count;

}

void out\_str(char \*str, int length, int number) { printf("%s\n", str); }

**Результаты:**



**Вывод**:

Была написана программа для сортировки и вывода строк, поддерживающая вывод такой дополнительной информации, как количество цифр в строке и количество перестановок строк при сортировке.