

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»
(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных
систем

Лабораторная работа №9

по дисциплине: Основы программирования
тема: «Использование структур»

Выполнил: ст. группы ПВ-201
Машуров Дмитрий Русланович

Проверил:
Притчин Иван Сергеевич
Брусенцева Валентина
Станиславовна

Белгород 2021 г.

Лабораторная работа № 9

Использование структур

Цель работы: получение навыков работы со структурами.

Задания для подготовки к работе

1. Изучить:
 - назначение структурного типа и правила его описания;
 - правила описания и инициализации переменных-структур;
 - назначение и описание битовых полей;
 - обращение к членам структур с помощью прямого и косвенного селектора.
2. Разработать алгоритм и составить программу для решения задачи соответствующего варианта.
3. Подобрать наборы тестовых данных.

Задание варианта №16

Даны адреса n человек ($n - \text{const}$). Вывести фамилии людей, живущих в разных городах по одинаковым адресам или сообщить, что таких людей не существует.

Выполнение:

1. Описание алгоритма и выделение подзадач:

Исходя из условия задачи, будем вводить данные пользователей по очереди и затем определять совпадение адресов и несовпадение городов

Выделение подзадач:

- 1) Определение равенства строк
- 2) Заполнение списка людей
- 3) Вывод людей, адреса которых совпадают, а города – нет

2. Структуры данных

Для определения данных о человеке:

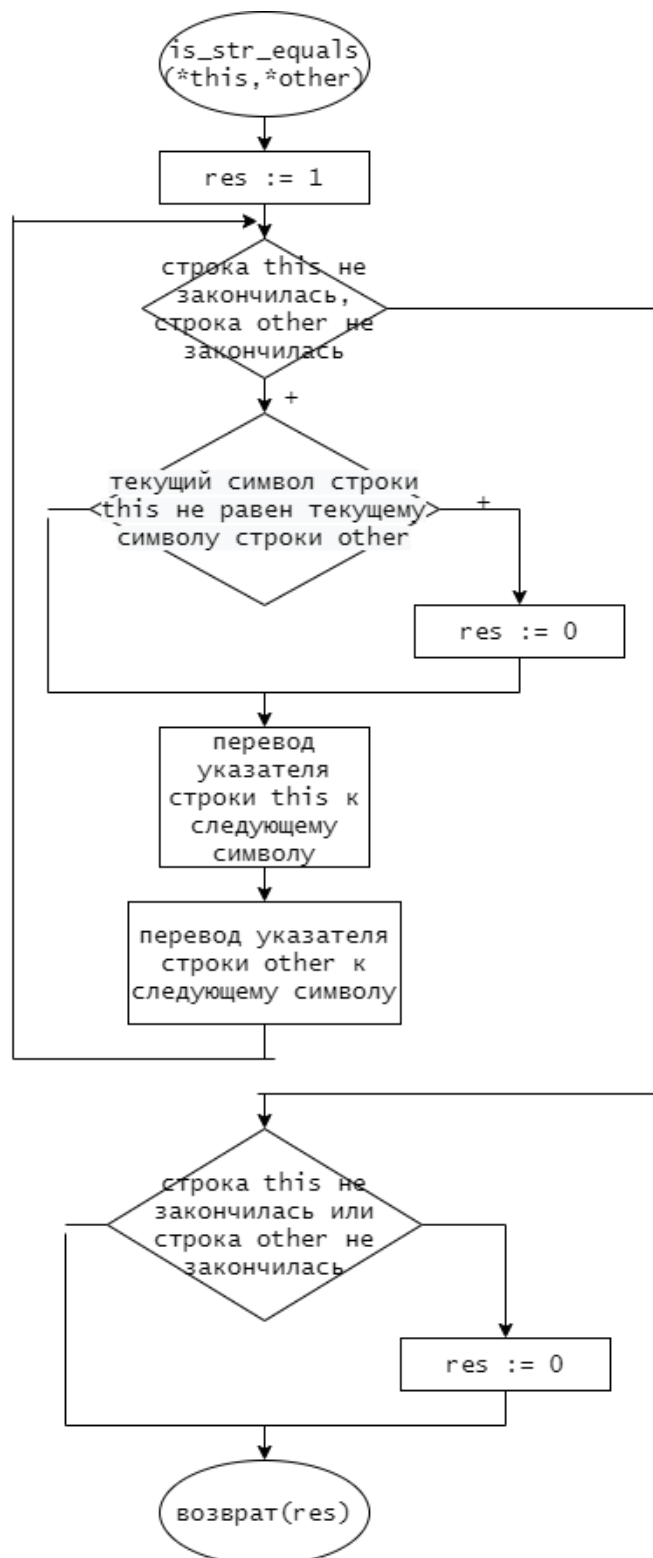
```
1) typedef struct data {  
    char second_name[50];  
    char city[50];  
    char address[50];  
} person;
```

Где `second_name` – фамилия, `city` – город, `address` – адрес

3. Описание подпрограмм

1) Определение равенства строк

- a. Заголовок: `int is_str_equals(const char *this, const char *other)`
- b. Назначение: возвращает «истину», если строка `this` равна строке `other`, иначе – «ложь»
- c. Блок-схема:

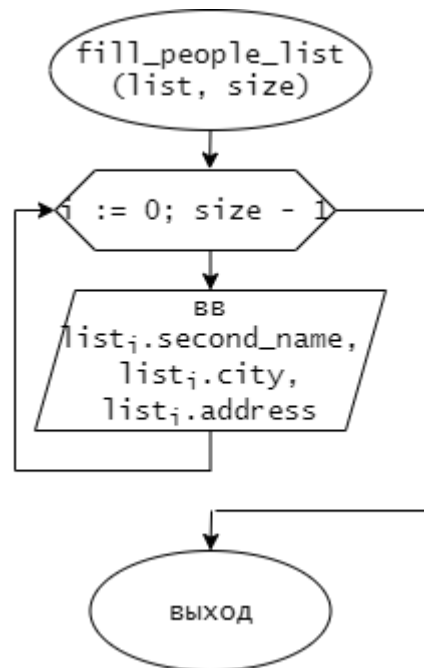


2) Заполнение списка людей

a. Заголовок: `void fill_people_list(person list[], int size)`

b. Назначение: заполняет список людей `list` размера `size`

c. Блок-схема:

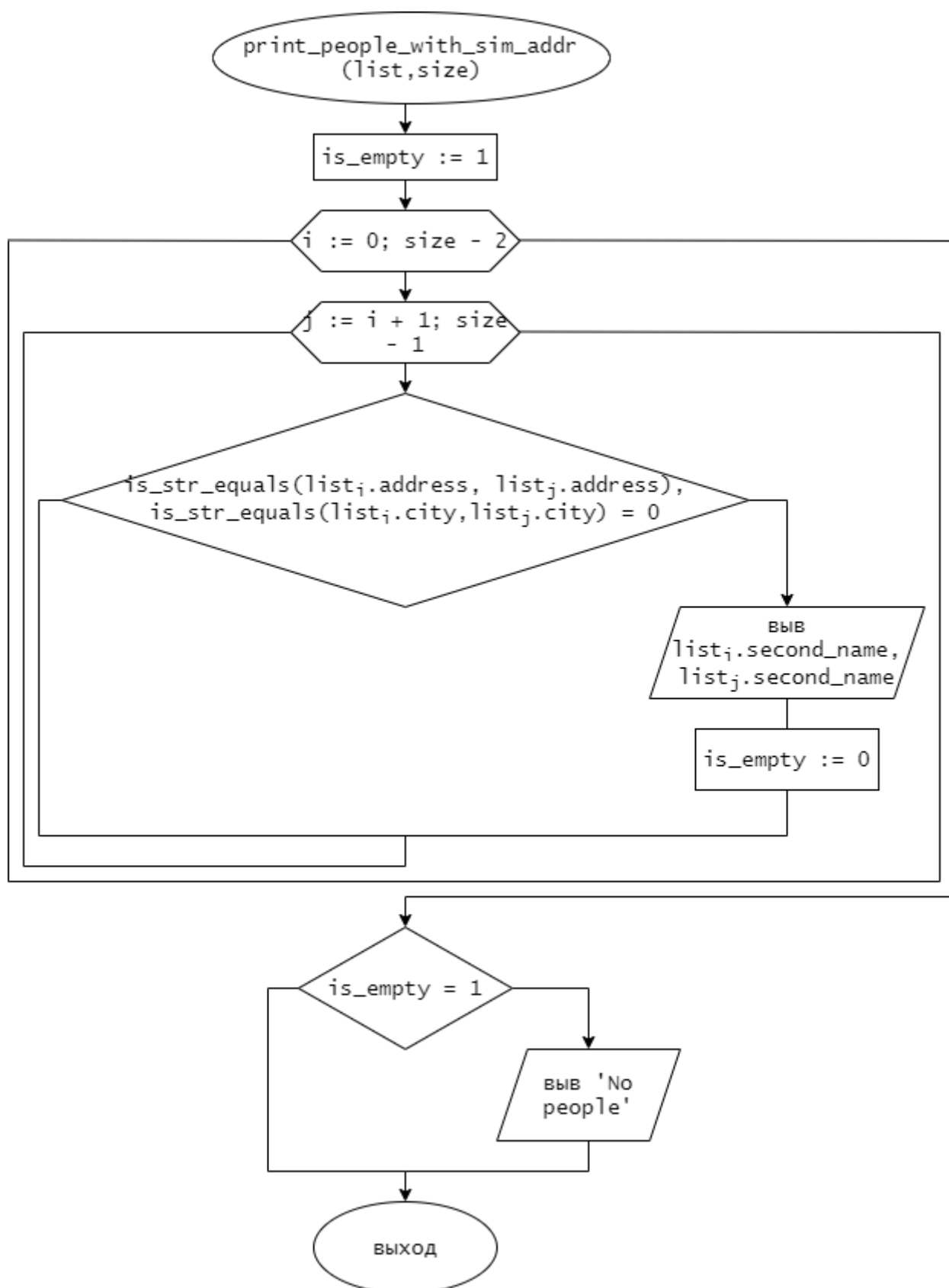


3) Вывод людей, адреса которых совпадают, а города – нет

a. Заголовок: `void print_people_with_sim_addr(person *list, int size)`

b. Назначение: вывод людей из списка `list` размера `size`, адреса которых совпадают, а города – нет

c. Блок-схема:



4. Тестовые данные

№	Ввод	Вывод
1	"Ivanov Moscow Lenina88" "Petrov Tula Ostrovskaya33a" "Sobolev Orel Lenina88"	"Ivanov --- Sobolev"
2	"Ivanov Belgorod Sumskaya8" "Petrov Belgorod Sumskaya9"	Нет таких людей

5. Текст программы

```
#include <stdio.h>
#include <stdlib.h>

typedef struct data {
    char second_name[50];
    char city[50];
    char address[50];
} person;

int is_str_equals(const char *this, const char *other) {
    while (*this != '\0' || *other != '\0') {
        if (*this != *other) {
            return 0;
        }
        this++; other++;
    }

    if (*other != '\0' || *this != '\0') {
        return 0;
    } else {
        return 1;
    }
}

void fill_people_list(person list[], int size) {
    for (int i = 0; i < size; ++i) {
        printf("Input: <Second name> <City> <Address>\n");
        scanf("%s %s %s", list[i].second_name, list[i].city,
list[i].address);
    }
}

void print_people_with_sim_addr(person *list, int size) {
    int is_empty = 1;
    for (int i = 0; i < size - 1; ++i) {
        for (int j = i + 1; j < size; ++j) {
            if ((is_str_equals(list[i].address, list[j].address)) &&
!(is_str_equals(list[i].city, list[j].city))) {
                printf("%s --- %s\n", list[i].second_name,
```

```

list[j].second_name);
        is_empty = 0;
    }
}

if (is_empty) {
    printf("No people\n");
}

}

int main() {
    int n;
    scanf("%d", &n);

    person people[n];
    fill_people_list(people, n);

    print_people_with_sim_addr(people, n);
}

```

6. Результаты:

Пример №1:

```

3
Input: <Second name> <City> <Address>
Ivanov Moscow Lenina88
Input: <Second name> <City> <Address>
Petrov Tula Ostrovskaya33a
Input: <Second name> <City> <Address>
Sobolev Orel Lenina88
Ivanov --- Sobolev

```

Пример №2:

```

2
Input: <Second name> <City> <Address>
Ivanov Belgorod Sumskaya8
Input: <Second name> <City> <Address>
Ivanov Belgorod Sumskaya9
No people

```

7. Анализ ошибок:

- Нет ошибок