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 №3**

по дисциплине: «Информатика»

«СТРУКТУРЫ В ЯЗЫКЕ СИ»

Вариант №13

Выполнили:Проверил:

Студенты гр. АВТ-419 Ядрышников О.Д.

Енина Ксения Сергеевна

Мерзляков Кирилл Андреевич

«06» ноября 2024 г. «\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г.

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

(оценка, подпись)

Новосибирск

2024

**Цель работы:** Ознакомиться с понятием структур. Научиться использовать структуру для организации простейших баз данных.

**Исходные данные и задание для варианта:**

4.1. Написать программу, работающую с базой данных в виде массива структур и выполняющую последовательный ввод данных в массив и последующую распечатку его содержимого. Состав структуры приведен в табл. 3.1. Типы данных выбрать самостоятельно.

4.2. Переписать программу п.4.1, используя массив указателей на структуры и динамическое выделение памяти. Выполнить сортировку массива. Способ сортировки массива приведен в табл. 3.1.

|  |  |  |
| --- | --- | --- |
| 13 | Наименование товара  Наименование заказчика  Количество заказанного товара  Фамилия менеджера | Расположить записи в массиве в алфавитном порядке по наименованию заказчика |

**Текст программы 4.1:**

#include <stdio.h>

#include <stdlib.h>

#define MAX\_STRING\_LEN 100

// Определение структуры для хранения данных о пользователе

typedef struct {

int count;

char name[MAX\_STRING\_LEN];

char customer\_name[MAX\_STRING\_LEN];

char manager\_surname[MAX\_STRING\_LEN];

} product;

int main() {

int products\_count;

printf("Введите количество товаров: ");

scanf("%d", &products\_count);

// Выделение памяти для массива структур

product \*products = (product \*)malloc(products\_count \* sizeof(product));

if (products == NULL) {

printf("Ошибка выделения памяти\n");

return 1;

}

// Последовательный ввод данных

for (int i = 0; i < products\_count; i++) {

printf("\nВведите название товара %d: ", i + 1);

scanf("%s", products[i].name);

printf("Введите количество товара %d: ", i + 1);

scanf("%d", &products[i].count);

printf("Введите наименование заказчика %d: ", i + 1);

scanf("%s", products[i].customer\_name);

printf("Введите фамилию менеджера %d: ", i + 1);

scanf("%s", products[i].manager\_surname);

}

// Распечатка содержимого массива

printf("\nТовары:\n");

for (int i = 0; i < products\_count; i++) {

printf("Название товара: %s, Наименование заказчика: %s, Количество товара: %d, Фамилия менеджера: %s\n",

products[i].name, products[i].customer\_name, products[i].count, products[i].manager\_surname);

}

// Освобождение памяти

free(products);

return 0;

}

**Текст программы 4.2:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAX\_STRING\_LEN 100

// Определение структуры для хранения данных о товаре

typedef struct {

int count;

char name[MAX\_STRING\_LEN];

char customer\_name[MAX\_STRING\_LEN];

char manager\_surname[MAX\_STRING\_LEN];

} product;

// Функция для сравнения товаров по наименованию заказчика (для сортировки)

int compareByCustomerName(const void \*a, const void \*b) {

const product \*prodA = \*(const product \*\*)a;

const product \*prodB = \*(const product \*\*)b;

return strcmp(prodA->customer\_name, prodB->customer\_name);

}

int main() {

int products\_count;

printf("Введите количество товаров: ");

scanf("%d", &products\_count);

// Выделение памяти для массива указателей на структуры

product \*\*products = (product \*\*)malloc(products\_count \* sizeof(product \*));

if (products == NULL) {

printf("Ошибка выделения памяти\n");

return 1;

}

// Последовательный ввод данных

for (int i = 0; i < products\_count; i++) {

products[i] = (product \*)malloc(sizeof(product));

if (products[i] == NULL) {

printf("Ошибка выделения памяти\n");

return 1;

}

printf("\nВведите название товара %d: ", i + 1);

scanf("%s", products[i]->name);

printf("Введите количество товара %d: ", i + 1);

scanf("%d", &products[i]->count);

printf("Введите наименование заказчика %d: ", i + 1);

scanf("%s", products[i]->customer\_name);

printf("Введите фамилию менеджера %d: ", i + 1);

scanf("%s", products[i]->manager\_surname);

}

// Сортировка массива по наименованию заказчика

qsort(products, products\_count, sizeof(product \*), compareByCustomerName);

// Распечатка содержимого массива

printf("\nОтсортированный список товаров по наименованию заказчика:\n");

for (int i = 0; i < products\_count; i++) {

printf("Название товара: %s, Наименование заказчика: %s, Количество товара: %d, Фамилия менеджера: %s\n",

products[i]->name, products[i]->customer\_name, products[i]->count, products[i]->manager\_surname);

}

// Освобождение памяти

for (int i = 0; i < products\_count; i++) {

free(products[i]);

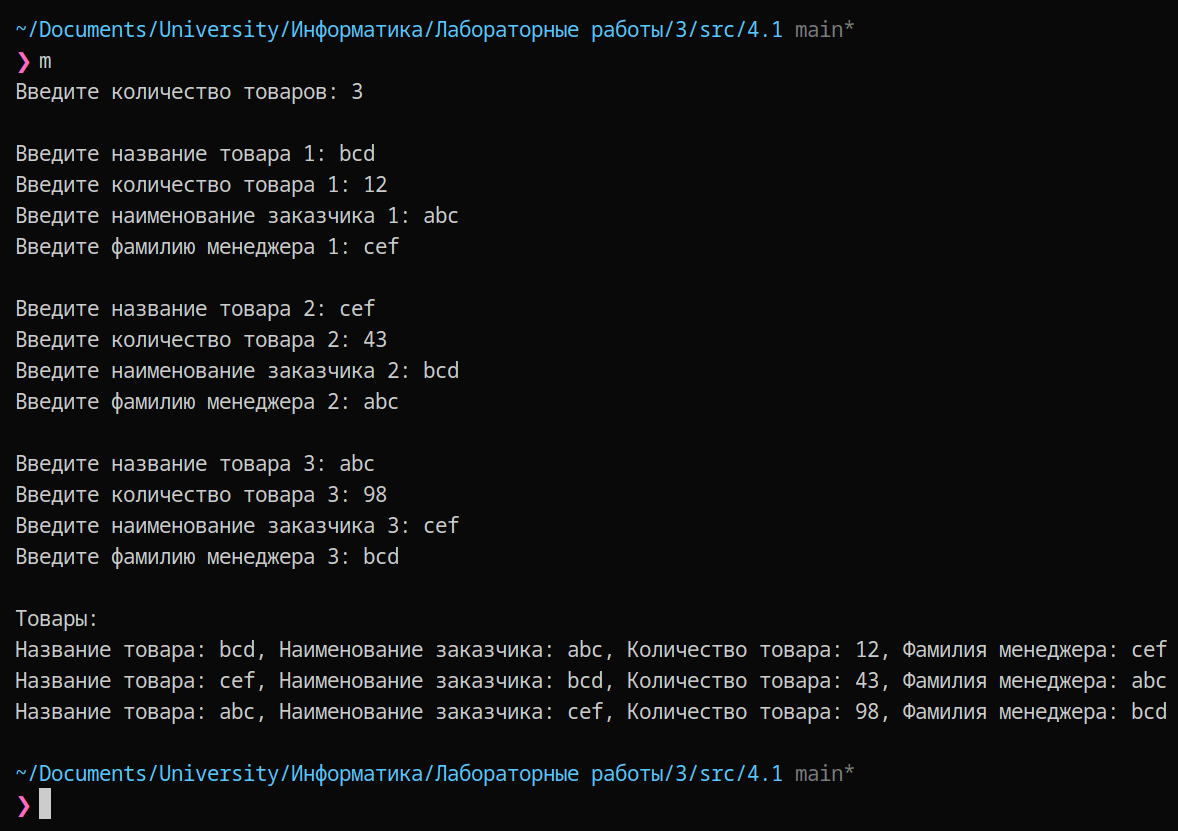
}

free(products);

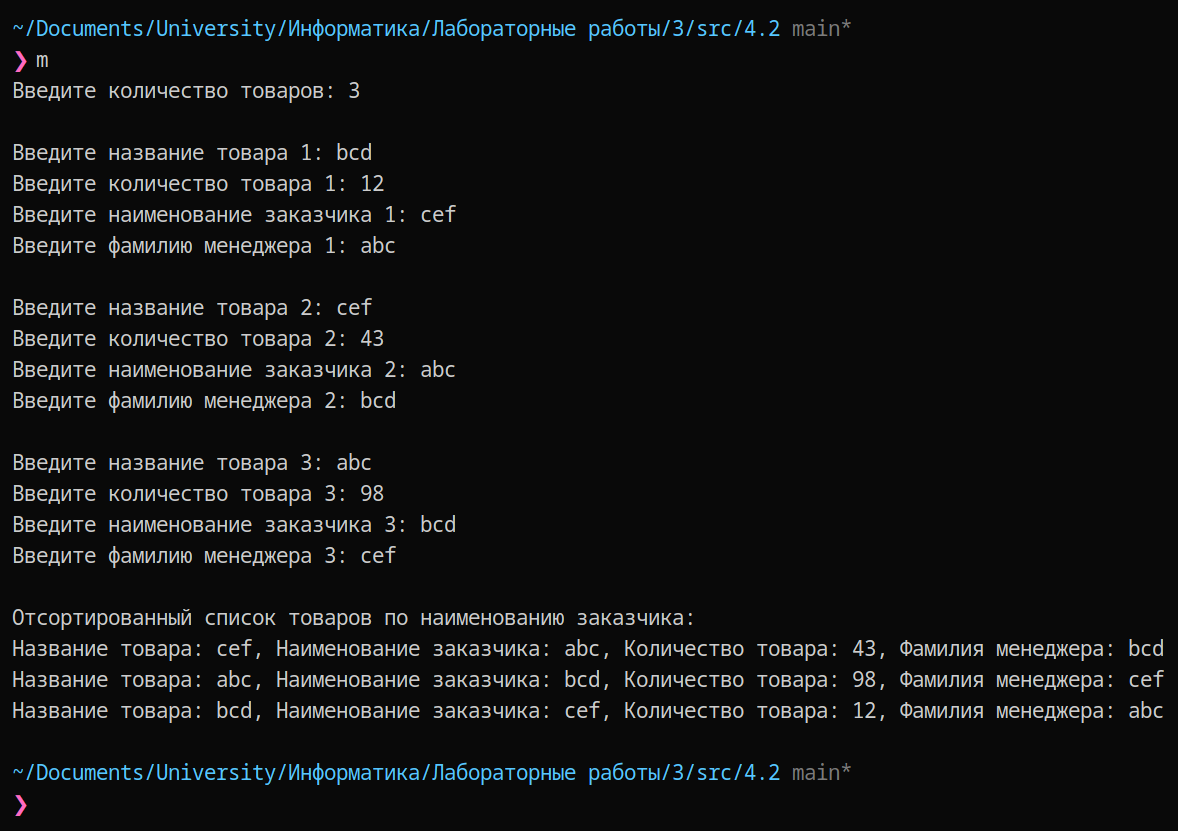
return 0;

}

**Результаты 4.1:**



**Результаты 4.2:**



**Вывод:** Мы освоили работу со структурами в языке программирования Си. Написали программу, в которой организуется простейшая база данных с помощью структур.