**Лабораторная работа № 3.**

СТРУКТУРЫ В ЯЗЫКЕ СИ

1. Цель работы

Ознакомиться с понятием структур. Научиться использовать структуру для организации простейших баз данных.

2. Краткие теоретические сведения

Структура в языке СИ объединяет логически связанные данные разных типов. Состав структуры определяется описанием шаблона:

***Struct ELEM // имя шаблона***

***{***

***int num; // названия элементов***

***char name[20];***

***float sum;***

***};***

Здесь ***ELEM*** -имя структуры; ***num***, ***name*** и ***sum*** – элементы структуры и их типы. В качестве элементов структуры могут быть также использованы имена других структур.

При описании структуры используется имя шаблона. Могут быть описаны отдельные структуры, массивы структур, а также указатели на структуры и массивы указателей. Примеры описаний:

***struct ELEM st; // структура***

***struct ELEM sm[10]; // массив структур***

***struct ELEM \*sp; // указатель на структуру***

***struct ELEM \*spm[10]; // массив указателей на структуры***

Для обращения к элементу структуры исполняется операция «**.**», например:

***st.num = 2;***

***sm[3].sum = 3.3;***

***scanf («%s», &sp.name);***

Для обращения к элементу структуры, адресуемой указателем, используется операция « ***->*** », например:

***sp->num = 2;***

***spm[3]->sum = 3.3;***

***spm->name = ’R’;***

3. Методические указания.

При написании программы следует использовать статические массивы структур или указателей на структуры. Размерности массивов – 3-4. Для динамического выделения памяти используйте функцию ***malloc()***. Для определения размера структуры в байтах удобно использовать операцию ***sizeof()***, возвращающую целую константу:

***struct ELEM \*sp;***

***sp = malloc(sizeof(structELEM));***

При выполнении п. 4.2. потребуется выполнять операцию перестановки элементов массива. Для этого необходимо описать дополнительный указатель на структуру. Можно также использовать вспомогательный массив указателей.

Ввод данных выполнить с помощью функций ***scanf()***.

4. Порядок выполнения работы

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.

Таблица 3.1

|  |  |  |
| --- | --- | --- |
| Вари ант | Структура данных | Задание п. 4.2. |
| 1 | Фамилия  Год рождения  Номер отдела  Оклад | Расположить записи в массиве в порядке возрастания (по году рождения) |
| 2 | Название детали  Год выпуска  Стоимость  Количество | Расположить записи в массиве в порядке возрастания стоимости |
| 3 | Название книги  Год издания  Количество страниц  Стоимость | Расположить записи в массиве в алфавитном порядке (по названию) |
| 4 | Фамилия  Дата рождения  Телефон  Сумма долга | Все записи с суммой долга, не равной 0, разместить в начале массива |
| 5 | Название покупки  Дата приобретения  Стоимость  Количество | Сгруппировать все записи по месяцам приобретения |
| 6 | Название команды  Игры  Очки  Сумма призового фонда | Расположить записи в порядке возрастания по сумме призового фонда |
| 7 | Фамилия  Группа  Номер в списке  Стипендия | Расположить записи в порядке возрастания номера в списке |
| 8 | Название маршрута  Протяженность  Количество остановок  Стоимость путевки | Расположить записи в порядке убывания стоимости |
| 9 | Фамилия спортсмена  Вид спорта  Количество медалей  Призовой фонд | Расположить в алфавитном порядке записи с ненулевым количеством медалей |
| 10 | Название детали  Количество  Вес  Адрес поставщика | Расположить записи в порядке возрастания количества деталей, с весом меньше заданного. |
| 11 | Фамилия  Группа  Место прохождения практики  Оценка | Расположить записи в массиве в порядке убывания оценки |
| 12 | Гос. номер  Марка/модель автомобиля  Фамилия владельца  Дата угона автомобиля | Расположить записи в массиве в порядке возрастания даты угона |
| 13 | Наименование товара  Наименование заказчика  Количество заказанного товара  Фамилия менеджера | Расположить записи в массиве в алфавитном порядке по наименованию заказчика |
| 14 | Номер рейса  Дата вылета  Фамилия пассажира  Номер кассы | Расположить записи в массиве в порядке возрастания номера кассы |
| 15 | Район города  Количество комнат в квартире  Площадь квартиры  Цена квартиры | Расположить записи в массиве в порядке возрастания количества комнат |
| 16 | Фамилия студента  Группа  Наименование дисциплины  Оценка | Расположить записи в массиве в порядке убывания оценки |
| 17 | Фамилия преподавателя  Наименование дисциплины  Номер курса  Аудитория | Расположить записи в массиве в алфавитном порядке по фамилии преподавателя |
| 18 | Учетная запись пользователя  Буква диска  Дисковая квота пользователя  Объем занятого дискового пространства | Расположить записи в массиве в алфавитном порядке по учетной записи пользователя |
| 19 | Название книги  Издательство  Цена книги  Количество книг | Расположить записи в массиве в алфавитном порядке по названию издательства |
| 20 | Наименование микросхемы  Тип корпуса  Наименование изготовителя  Цена | Расположить записи в массиве в порядке возрастания цены |

5. Содержание отчета

5.1. Титульный лист

5.2. Цель работы.

5.3. Задание.

5.4. Текст(ы) программ.

5.5. Полученные результаты.

5.6. Выводы.

6. Контрольные вопросы

6.1. Правила описания структур.

6.2. Операции, используемые для выделения элементов структур.

6.3. Выделение элементов структур, адресуемых указателем.

6.4. Назначение функции ***malloc()****.*

6.5. Отличие ***calloc()*** от ***malloc()***.

6.6. Правила инициализации структур при описании.

6.7. Особенности выделения памяти под структуры.

6.8. В каком случае размерность массива при его описании можно не указывать?

6.9. Назначение функций ***realloc()*** и ***free()***.

6.10. Как передать структуру в функцию?