

Лабораторная работа 13. Типы данных, определяемые пользователем

1. ЦЕЛЬ РАБОТЫ

Изучение возможности переименования типов, способов объявления и использования перечислимых типов, структур и объединений в языке C++.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

Встроенные типы данных, указатели и массивы образуют основу для представления и обработки информации на языке C++. Кроме того, язык C++ позволяет программисту самому определять наиболее подходящие для конкретной задачи типы, способные адекватно представлять сложно структурированные данные реального мира.

Переименование типов. Любому типу в C++ можно присвоить простое имя или переименовать его. Это делается с помощью ключевого слова `typedef`:

```
typedef тип новое_имя_типа;
```

или

```
typedef тип новое_имя_типа [размер_массива][...];
```

для типов-массивов. Кроме того, можно вводить имена для типов указателей на функцию и т. п. Примеры:

```
typedef short ArrIndex; // Псевдоним для short.  
typedef char MessageStr[80]; // Имя типа для массивов char[80].  
typedef int *IPtrFunc(void); // Функция, возвращающая указатель  
                             // на int.  
typedef int (*IFuncPtr)(void); // Указатель на функцию,  
                             // возвращающую int.
```

В общем, `typedef` является средством упрощения записи операторов объявления переменных.

Перечислимые типы. Ключевое слово `enum` позволяет описать перечислимый тип, представляющий переменные, которые могут принимать значения из заданного набора целых именованных констант. Определение перечислимого типа выглядит так:

```
enum имя-этикетка {имя_константы [= значение], ...};
```

Значение равно по умолчанию нулю для первого из перечислителей (так обычно называют определяемые в `enum` константы). Любая другая константа, для которой значение не указано, принимается равной значению предыдущей константы плюс единица. Например:

```
enum Status { Success = 1,  
             Wait, Proceed,  
             Error = -1  
};
```

В операторе `enum` после закрывающей фигурной скобки можно сразу объявить несколько переменных данного типа, например, так:

```
enum Status Proc1Status, Proc2Status;
```

При желании можно воспользоваться ключевым словом `typedef` и ввести для перечисления подлинное новое имя. Предыдущее объявление можно переписать так:

```
typedef enum _Status {Success = 1,  
                      Wait, Proceed,  
                      Error = -1 } Status;
```

В этом случае `Status` будет полноценным именем перечислимого типа.

Структуры. Массивы позволяют обращаться с набором логически связанных однотипных элементов как с единым целым. Если же требуется хранить набор разнородных, но логически связанных данных, описывающих, например, состояние некоторого объекта реального мира, используются структуры. Синтаксис структуры имеет такой вид:

```
struct Имя_типа {определения_элементов} [переменные];
```

Вот пример структуры, предназначенной для хранения основных сведений о человеке:

```
struct Person  
{char lastName[32]; // Фамилия.  
 char firstName[32]; // Имя.  
 short age; // Возраст.  
 long phoneNum; // Телефон как длинное целое.  
 } aPerson; // Объявляем переменную типа struct Person.
```

Данная структура группирует различные данные, относящиеся к конкретному человеку. Как и в случае перечислений, в определении структуры можно сразу объявить переменные структурного типа, указав их имена после закрывающей фигурной скобки. После того, как структура объявлена, имя структурного типа можно назначить переменным:

```
struct Person P1, P2;
```

Правила C++ разрешают ключевое слово `struct` опускать и писать:

```
Person P1, P2;
```

Для доступа к отдельным элементам структуры (полям) имеются две операции: точка и стрелка, за которыми следует имя элемента. С именем переменной применяется точка, с указателем — стрелка. Имея в виду предыдущее определение, можно было бы написать:

```
P1.age=20;
```

Значения полей структуры можно определять вводом, присваиванием, инициализацией. Пример инициализации в объявлении:

```
Person P1={"Иванов", "Петр", 20, 2562525};
```

В программе можно объявить указатель на структуру:

```
Person *pntP1, P1;
```

Тогда после выполнения оператора присваивания: `pntP1=&P1;`

к каждому элементу структурной переменной можно обращаться тремя способами:

`P1.lastName` или `(*pntP1).lastName` или `pntP1->lastName`

Битовые поля. В качестве элементов структуры можно определять битовые поля. Для них задается ширина поля в битах, и компилятор отводит под элемент ровно столько бит, сколько указано. Несколько битовых полей может быть таким образом упаковано в одном слове. Синтаксис битового поля:

тип [имя поля]: ширина поля;

Тип поля может быть `int` или `unsigned int`. Доступ к битовым полям осуществляется так же, как и к регулярным элементам структуры. Если имя_поля отсутствует, место под поле отводится, но оно остается недоступным. Это будут просто “заполняющие” биты. Пример описания структуры, в битовое поле имеет минимальную величину – 1 бит:

```
struct onebit {unsigned bit:1;  
               } cod;
```

Для переменной `cod` в памяти будет выделено одно слово, но использоваться будет только один бит.

Битовые поля применяются либо там, где необходима плотная упаковка информации (как это бывает при передаче функции некоторого набора логических флагов), либо, например, для отображения регистров внешнего устройства, которые часто бывают организованы как совокупность небольших полей и отдельных битов.

Объединения. Объединения, определяемые с помощью ключевого слова `union`, похожи по своему виду на структуры:

```
union имя_объединения {список_элементов} [переменные];
```

Отличие состоит в том, что все элементы объединения занимают одно и то же место в памяти, они перекрываются. Компилятор отводит под объединение память, достаточную для размещения наибольшего элемента.

Объединения полезны, когда требуется обеспечить своего рода “полиморфное поведение” некоторого объекта. Доступ к элементам объединения (их иногда называют разделами) осуществляется так же, как и в структурах, — посредством точки или стрелки, за которыми следует имя раздела. Пример использования объединений см. в лекциях.

3. ЗАДАНИЕ НА РАБОТУ

1. Разработать программу со структурой `student` (см. п. Порядок выполнения работы).
2. Разработать самостоятельно приложения для решения задачи по своему варианту.

4. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Задание 1. Ознакомиться с теоретическим материалом, приведенным в пункте «Краткие теоретические положения» данных методических указаний, а также с конспектом лекций и рекомендуемой литературой по данной теме.

Задание 2. Разработка программы с использованием структур.

Разработаем программу, в которой будет храниться и обрабатываться информация о стипендии студентов разных групп. Информацию о студентах требуется организовать в виде структуры такого вида:



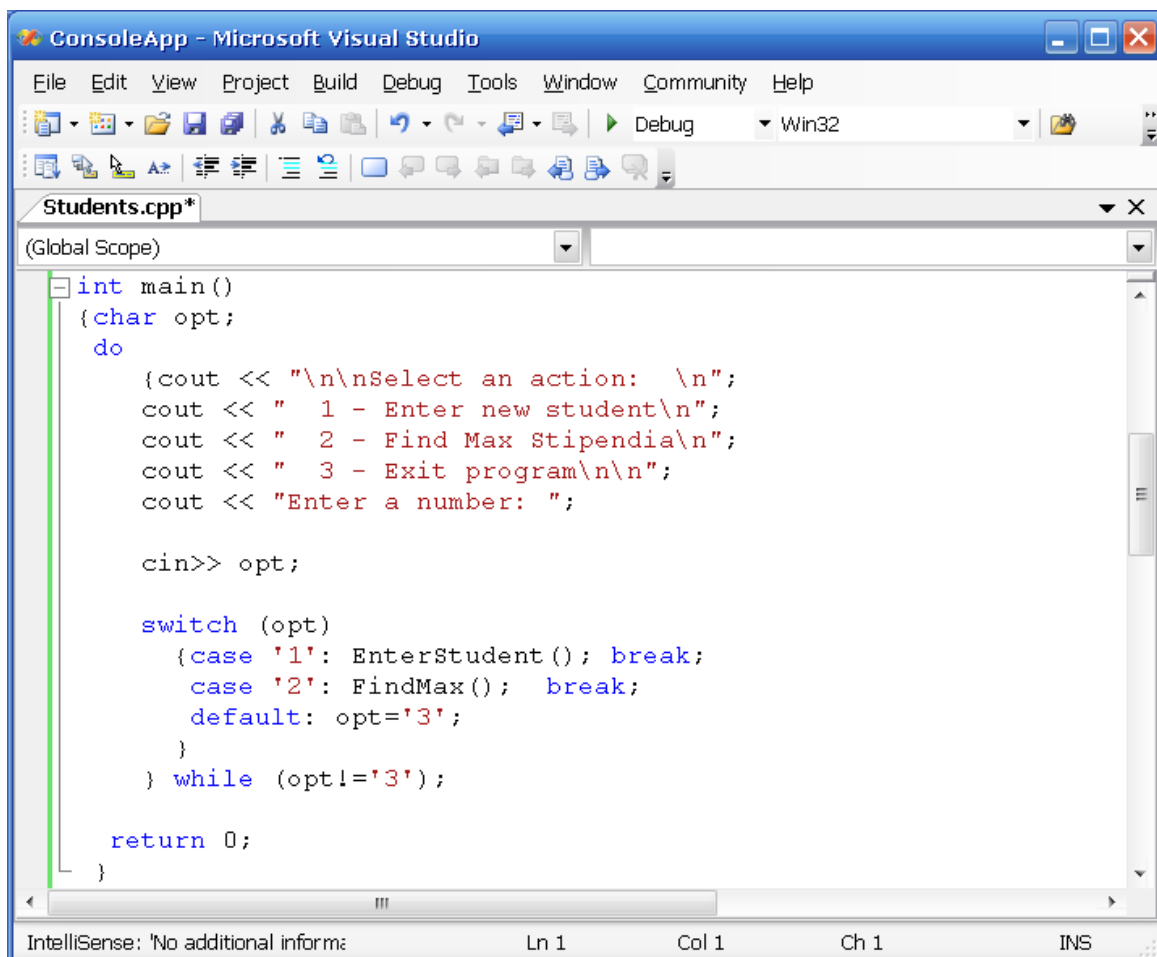
Объявление структурного типа может быть следующим:

```
struct student {char fam[20];  
                int grup;  
                int stip;  
            }
```

После создания пустого проекта **Students** в него необходимо добавить исходные файлы программы. В окне кода вначале сделаем необходимые глобальные объявления:

```
#include <iostream>  
#include <conio.h>  
  
using namespace std;  
  
//объявление структуры для хранения данных об одном студенте  
struct student {  
    char fam[20];  
    int grup;  
    int stip;  
};  
  
//объявим массив для хранения информации о 30 студентах  
student stud[30];  
int i; //счетчик для подсчета введенных записей  
//-----  
  
//объявление функций для ввода данных о студенте  
//и поиске максимальной стипендии  
void EnterStudent();  
void FindMax();
```

Затем введем код функции `main()`:



```

ConsoleApp - Microsoft Visual Studio
File Edit View Project Build Debug Tools Window Community Help
Students.cpp*
(Global Scope)
int main()
{
    char opt;
    do
    {
        cout << "\n\nSelect an action: \n";
        cout << " 1 - Enter new student\n";
        cout << " 2 - Find Max Stipendia\n";
        cout << " 3 - Exit program\n\n";
        cout << "Enter a number: ";

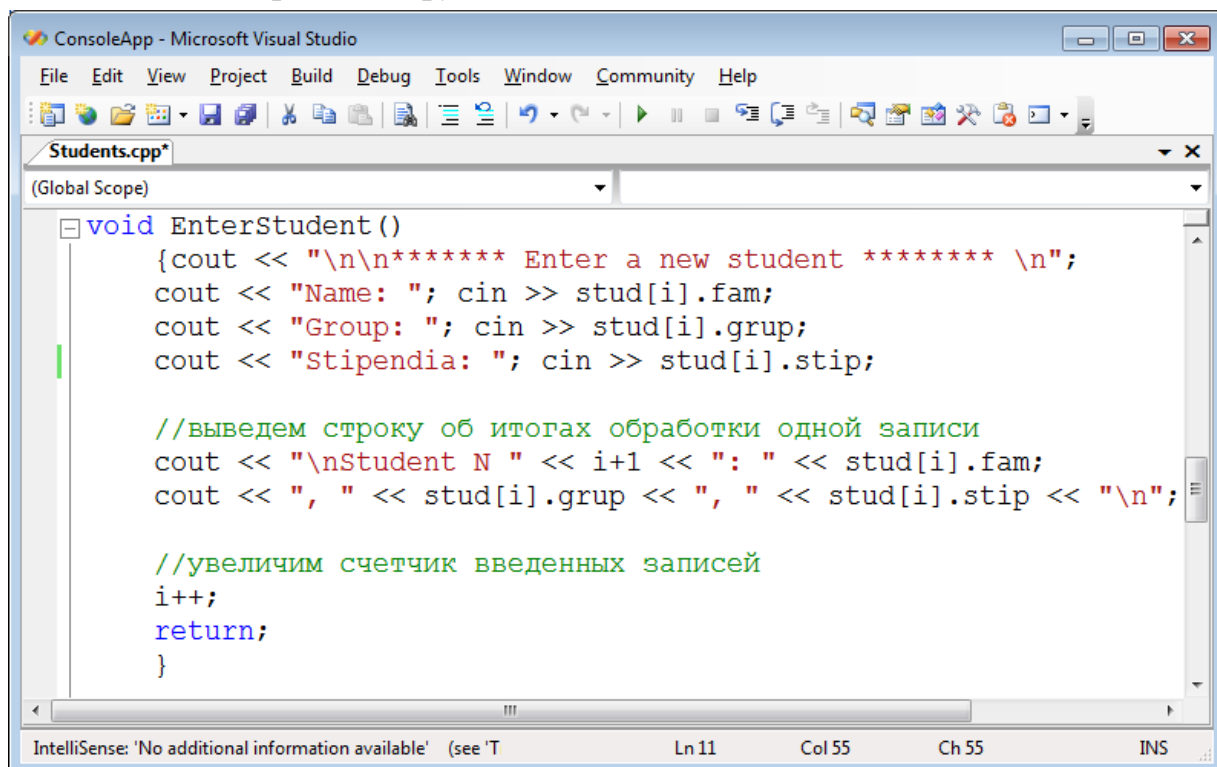
        cin >> opt;

        switch (opt)
        {
            case '1': EnterStudent(); break;
            case '2': FindMax(); break;
            default: opt = '3';
        }
    } while (opt != '3');

    return 0;
}
IntelliSense: 'No additional information available' Ln 1 Col 1 Ch 1 INS

```

После этого определим функцию EnterStudent():



```

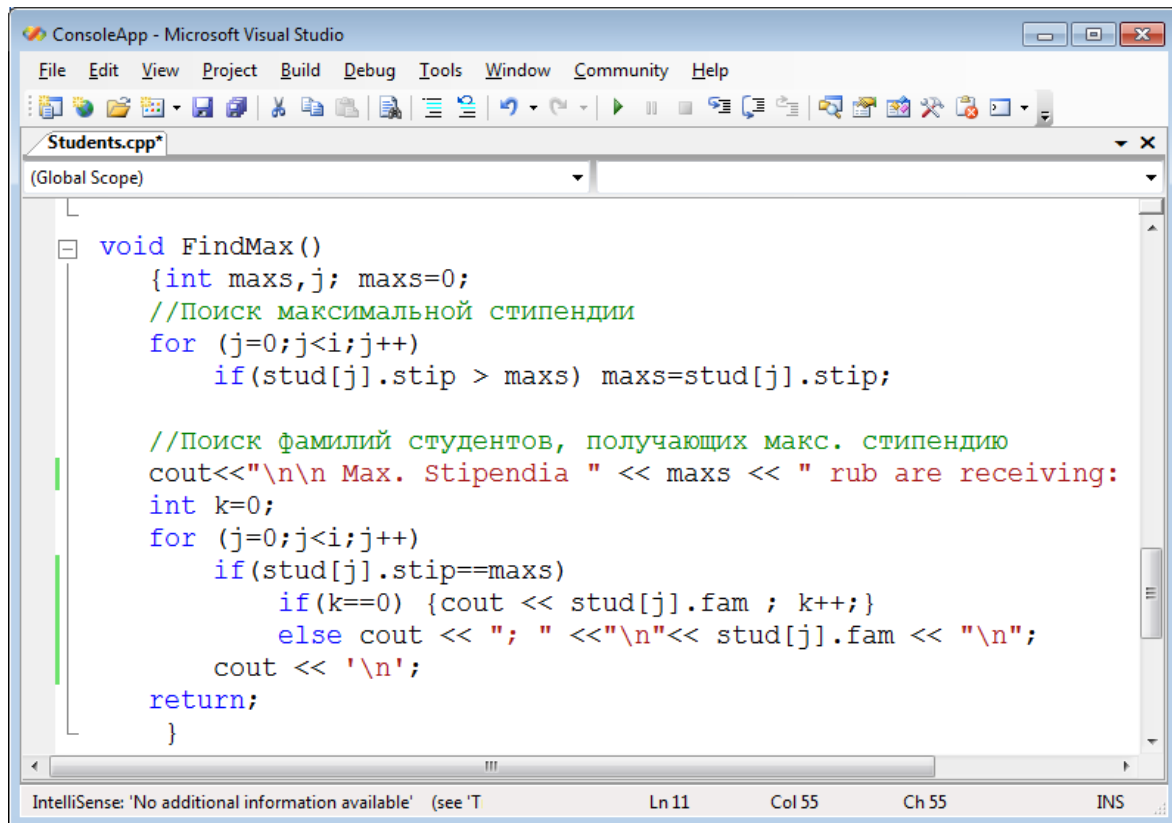
ConsoleApp - Microsoft Visual Studio
File Edit View Project Build Debug Tools Window Community Help
Students.cpp*
(Global Scope)
void EnterStudent()
{
    cout << "\n\n***** Enter a new student ***** \n";
    cout << "Name: "; cin >> stud[i].fam;
    cout << "Group: "; cin >> stud[i].grup;
    cout << "Stipendia: "; cin >> stud[i].stip;

    //выведем строку об итогах обработки одной записи
    cout << "\nStudent N " << i+1 << ": " << stud[i].fam;
    cout << ", " << stud[i].grup << ", " << stud[i].stip << "\n";

    //увеличим счетчик введенных записей
    i++;
    return;
}
IntelliSense: 'No additional information available' (see 'T Ln 11 Col 55 Ch 55 INS

```

А затем определим функцию для поиска студентов, получающих максимальную стипендию `FindMax()`, например, так:



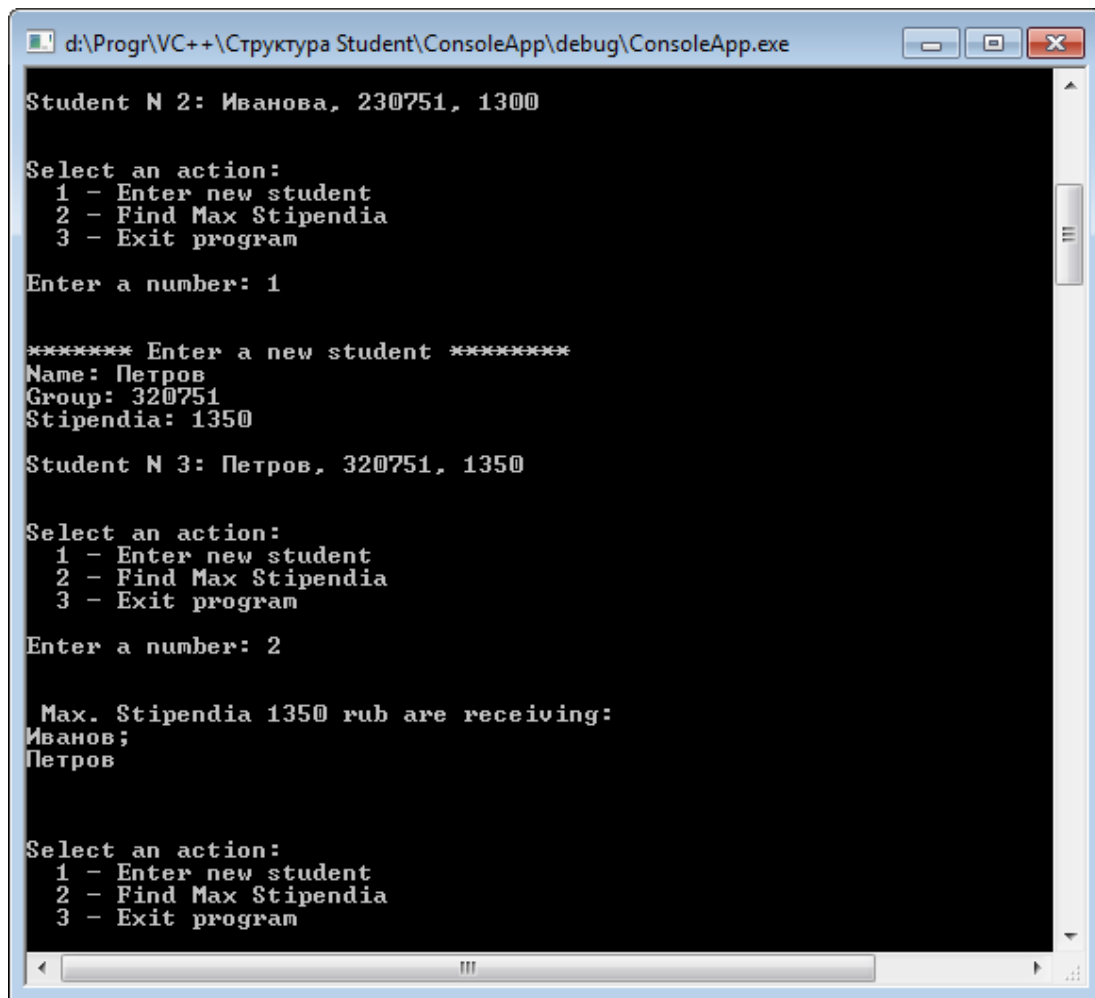
The screenshot shows the Microsoft Visual Studio IDE with a C++ project named 'ConsoleApp'. The file 'Students.cpp' is open, showing the 'Global Scope'. The code defines a function `FindMax()` that finds the maximum stipend and lists the students receiving it.

```
void FindMax()
{
    int maxs, j; maxs=0;
    //Поиск максимальной стипендии
    for (j=0; j<i; j++)
        if (stud[j].stip > maxs) maxs=stud[j].stip;

    //Поиск фамилий студентов, получающих макс. стипендию
    cout<<"\n\n Max. Stipendia " << maxs << " rub are receiving:
    int k=0;
    for (j=0; j<i; j++)
        if (stud[j].stip==maxs)
            if (k==0) {cout << stud[j].fam ; k++;}
            else cout << "; " <<"\n"<< stud[j].fam << "\n";
    cout << '\n';
    return;
}
```

The status bar at the bottom indicates 'IntelliSense: 'No additional information available' (see 'T', Ln 11, Col 55, Ch 55, INS).

Сохраните проект и запустите его. Проверьте работу программы при вводе данных и поиске максимальной стипендии. Примерный результат работы программы представлен на рисунке.



```
d:\Progr\VC++\Структура Student\ConsoleApp\debug\ConsoleApp.exe

Student N 2: Иванова, 230751, 1300

Select an action:
1 - Enter new student
2 - Find Max Stipendia
3 - Exit program

Enter a number: 1

***** Enter a new student *****
Name: Петров
Group: 320751
Stipendia: 1350

Student N 3: Петров, 320751, 1350

Select an action:
1 - Enter new student
2 - Find Max Stipendia
3 - Exit program

Enter a number: 2

Max. Stipendia 1350 rub are receiving:
Иванов;
Петров

Select an action:
1 - Enter new student
2 - Find Max Stipendia
3 - Exit program
```

Задание 3. Разработайте программу по своему варианту.

Покажите результаты работы преподавателю. Оформите отчет по работе.

5. ОФОРМЛЕНИЕ ОТЧЕТА

Отчет по работе должен содержать:

- название и цель работы;
- номер варианта;
- для программы по своему варианту – текст задачи, **описание разработанных структур данных и функций**, текст кода программы, результаты работы программы.

6. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Шилдт Г. С++: базовый курс, 3-е издание. : Пер. с англ. – М.: «Издательский дом «Вильямс», 2005. – 624 с.
2. Пахомов Б.И. С/C++ и MS Visual C++ для начинающих. – СПб.: БХВ-Петербург, 2008. – 624 с.

7. Варианты заданий

Вар.	Задание												
1	Объявить структуру для хранения библиотечной карточки: шифр, автор, наименование книги, количество страниц, год издания, издательство. Объявить массив карточек и поместить в него сведения о нескольких книгах. Разработать программу, позволяющую выполнить создание новой карточки, запись всех карточек в файл, чтение карточек из файла, выдачу информации о книге (книгах) по фамилии автора или году издания, а также упорядочения информации по названию фамилии автора или году издания.												
2	Объявить структуру для хранения информации о знакомых: фамилия, имя, отчество, номер телефона, день рождения, адрес. Объявить массив «Записная книжка» и поместить в него сведения о нескольких людях. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение записной книжки из файла, выдачу информации о человеке по номеру телефона, фамилии или дню рождения, а также упорядочения информации по названию фамилии или номеру телефона (в алфавитном порядке).												
3	Объявить структуру для хранения информации о деталях: наименование, масса, габаритные размеры, материал, количество на складе. Объявить массив деталей и поместить в него сведения о нескольких изделиях. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о деталях из файла, выдачу информации о детали по номеру наименованию, подсчет массы всех деталей.												
4	<div>Сформировать структуру «заказ» следующей формы:</div> <table><tr><td colspan="3">Дата</td><td rowspan="2">Номер заказа</td><td colspan="2">Телефон</td><td rowspan="2">Город</td></tr><tr><td>Число</td><td>Месяц</td><td>год</td><td>основной</td><td>дополнительный</td></tr></table> <div>Объявить массив заказов и поместить в него сведения о нескольких заказах. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о заказах из файла, выдачу информации о заказе по номеру заказа, подсчет числа заказов за некоторый день, месяц или год.</div>	Дата			Номер заказа	Телефон		Город	Число	Месяц	год	основной	дополнительный
Дата			Номер заказа	Телефон		Город							
Число	Месяц	год		основной	дополнительный								
5	Объявить структуру для хранения информации о выполнении плана: наименование изделия, код изделия, единица измерения, план выпуска, фактически выпущено. Объявить массив «Результаты выполнения плана» и поместить в него сведения о нескольких изделиях. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение данных из файла, выдачу информации по введенному коду изделия, подсчет суммарного отклонения от плана (недовыполнение, перевыполнение) для изделий, имеющих введенные единицы измерения, а также упорядочения изделий по наименованию (в алфавитном порядке).												
6	<div>Сформировать структуру «Товар» следующей формы:</div> <table><tr><td rowspan="2">Код товара</td><td rowspan="2">Наименование товара</td><td rowspan="2">Количество, шт</td><td colspan="2">Стоимость, руб</td></tr><tr><td>за штуку</td><td>всего</td></tr></table>	Код товара	Наименование товара	Количество, шт	Стоимость, руб		за штуку	всего					
Код товара	Наименование товара				Количество, шт	Стоимость, руб							
		за штуку	всего										

	Объявить массив товаров на складе и поместить в него сведения о нескольких товарах. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о товарах из файла, выдачу информации о товаре по первым буквам наименования, подсчет числа общей стоимости товаров на складе, а также упорядочения товаров по наименованию (в алфавитном порядке).										
7	Объявить структуру для хранения информации о результатах сдачи экзамена: номер зачетки, фамилия студента, название экзамена, оценка (неуд., уд., хор., отл.). Объявить массив «Табель успеваемости группы» и поместить в него сведения о группе студентов. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение табеля из файла, выдачу информации о студентах по зачетной книжке, подсчет среднего балла для выбранного студента, подсчет количества отличников и двоечников в группе, а также упорядочение информации по фамилии или номеру зачетки.										
8	<div>Сформировать структуру «Товар» следующей формы:</div> <table><tr><td rowspan="2">Код товара</td><td rowspan="2">Наименование товара</td><td rowspan="2">Количество, шт</td><td colspan="2">Стоимость, руб</td></tr><tr><td>за штуку</td><td>всего</td></tr></table> <div>Объявить массив товаров на складе и поместить в него сведения о нескольких товарах. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о товарах из файла, выдачу информации о товаре по первым буквам наименования, подсчет числа общей стоимости товаров на складе.</div>	Код товара	Наименование товара	Количество, шт	Стоимость, руб		за штуку	всего			
Код товара	Наименование товара				Количество, шт	Стоимость, руб					
		за штуку	всего								
9	Объявить структуру для хранения информации о спортсмене: фамилия, время в беге на 100 м., время в беге на 3 км., результаты прыжков в высоту и в длину. Объявить массив «Результаты соревнований» и поместить в него сведения о нескольких спортсменах. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о результатах соревнований из файла, выдачу информации о спортсмене по какому-либо результату, определение лучшего спортсмена по введенному виду спорта.										
10	Объявить структуру для хранения информации о результатах сдачи экзамена: номер зачетки, фамилия студента, название экзамена, оценка (неуд., уд., хор., отл.). Объявить массив «Табель успеваемости группы» и поместить в него сведения о группе студентов. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение табеля из файла, выдачу информации о студентах по зачетной книжке, подсчет среднего балла для выбранного студента, подсчет количества отличников в группе, а также упорядочение информации по фамилии или экзамену.										
11	<div>Сформировать структуру «Изделие» следующей формы:</div> <table><tr><td rowspan="2">№ п/п</td><td rowspan="2">Инвентарный номер</td><td colspan="3">Дата поступления</td><td rowspan="2">Наименование</td><td rowspan="2">Количество</td></tr><tr><td>Число</td><td>Месяц</td><td>Год</td></tr></table> <div>Объявить массив «Инвентарная ведомость» и поместить в него сведения о нескольких изделиях на складе. Разработать программу, позволяющую</div>	№ п/п	Инвентарный номер	Дата поступления			Наименование	Количество	Число	Месяц	Год
№ п/п	Инвентарный номер			Дата поступления					Наименование	Количество	
		Число	Месяц	Год							

	выполнить создание новой записи, запись всех данных в файл, чтение информации об изделиях из файла, выдачу информации об изделии по инвентарному номеру, по первым буквам наименования, подсчет общего количества изделий, поступивших на склад в определенный месяц года, а также упорядочения товаров по наименованию (в алфавитном порядке).										
12	Объявить структуру для хранения информации о результатах сдачи экзамена: номер группы, фамилия студента, название экзамена, оценка (неуд., уд., хор., отл.). Объявить массив «Табель успеваемости» и поместить в него сведения о студентах из разных групп. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение табеля из файла, выдачу информации о студентах по фамилии, номеру группы, подсчет среднего балла по выбранному предмету.										
13	Объявить структуру для хранения информации о работниках: отдел, ФИО, домашний адрес, телефон. Объявить массив для хранения данных о работниках и поместить в него сведения о нескольких людях. Разработать программу, позволяющую выполнить создание записи о новом работнике, запись всех данных в файл, чтение данных из файла, выдачу информации о работниках по фамилии, улице (из поля адреса); подсчет количества работников в каждом отделе; вывод всех данных, хранящихся в файле, а также упорядочение информации о сотрудниках по фамилии или названию отдела.										
14	<p>Сформировать структуру «Изделие» следующей формы:</p> <table><tr><td rowspan="2">№ п/п</td><td rowspan="2">Инвентарный номер</td><td colspan="3">Дата поступления</td><td rowspan="2">Наименование</td><td rowspan="2">Количество</td></tr><tr><td>Число</td><td>Месяц</td><td>Год</td></tr></table> <p>Объявить массив «Инвентарная ведомость» и поместить в него сведения о нескольких изделиях на складе. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации об изделиях из файла, выдачу информации об изделии по инвентарному номеру, по первым буквам наименования, подсчет общего количества изделий, поступивших на склад в определенный месяц года.</p>	№ п/п	Инвентарный номер	Дата поступления			Наименование	Количество	Число	Месяц	Год
№ п/п	Инвентарный номер			Дата поступления					Наименование	Количество	
		Число	Месяц	Год							
15	Объявить структуру для хранения информации о результатах сдачи экзамена: номер группы, фамилия студента, название экзамена, оценка (неуд., уд., хор., отл.). Объявить массив «Табель успеваемости» и поместить в него сведения о студентах из разных групп. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение табеля из файла, выдачу информации о студентах по фамилии, номеру группы, подсчет среднего балла по выбранному предмету и по выбранной группе.										
16	Объявить структуру для хранения информации о поставщиках товара: наименование поставщика, адрес, телефон, наименование поставленного товара, дата поставки. Объявить массив для хранения данных о поставщиках и поместить в него сведения о нескольких поставщиках и поставленных ими товарах. Разработать программу, позволяющую выполнить создание записи о новом поставщике (новой поставке товара), запись всех данных в файл, чтение данных из файла, выдачу информации о поставщике по наименованию, городу (из поля адреса); определение даты последней поставки какого-либо товара;										

	вывод всех данных.												
17	Объявить структуру для хранения информации о знакомых: фамилия, имя, отчество, номер телефона, день рождения, адрес. Объявить массив «Записная книжка» и поместить в него сведения о нескольких людях. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение записной книжки из файла, выдачу информации о человеке по номеру телефона, фамилии или дню рождения.												
18	Объявить структуру для хранения информации о деталях: наименование, масса, габаритные размеры, материал, количество на складе. Объявить массив деталей и поместить в него сведения о нескольких изделиях. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о деталях из файла, выдачу информации о детали по номеру или наименованию, подсчет массы всех деталей.												
19	<p>Сформировать структуру «заказ» следующей формы:</p> <table><tr><td colspan="3">Дата</td><td rowspan="2">Номер заказа</td><td colspan="2">Телефон</td><td rowspan="2">Город</td></tr><tr><td>Число</td><td>Месяц</td><td>год</td><td>основной</td><td>дополнительный</td></tr></table> <p>Объявить массив заказов и поместить в него сведения о нескольких заказах. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о заказах из файла, выдачу информации о заказе по номеру заказа, подсчет числа заказов за некоторый месяц или за год.</p>	Дата			Номер заказа	Телефон		Город	Число	Месяц	год	основной	дополнительный
Дата			Номер заказа	Телефон		Город							
Число	Месяц	год		основной	дополнительный								
20	Объявить структуру для хранения информации о выполнении плана: наименование изделия, код изделия, единица измерения, план выпуска, фактически выпущено. Объявить массив «Результаты выполнения плана» и поместить в него сведения о нескольких изделиях. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение данных из файла, выдачу информации по введенному коду изделия, подсчет суммарного отклонения от плана (недовыполнение, перевыполнение) для изделий, имеющих введенные единицы измерения или наименование.												
21	<p>Сформировать структуру «заказ» следующей формы:</p> <table><tr><td colspan="3">Дата</td><td rowspan="2">Номер заказа</td><td colspan="2">Телефон</td><td rowspan="2">Город</td></tr><tr><td>Число</td><td>Месяц</td><td>год</td><td>основной</td><td>дополнительный</td></tr></table> <p>Объявить массив заказов и поместить в него сведения о нескольких заказах. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о заказах из файла, выдачу информации о заказе по номеру заказа, подсчет числа заказов за некоторый год или из некоторого города.</p>	Дата			Номер заказа	Телефон		Город	Число	Месяц	год	основной	дополнительный
Дата			Номер заказа	Телефон		Город							
Число	Месяц	год		основной	дополнительный								
22	Объявить структуру для хранения информации о результатах сдачи экзамена: номер группы, фамилия студента, название экзамена, оценка (от 0 до 100 баллов). Объявить массив «Табель успеваемости» и поместить в него сведения о студентах из разных групп. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение табеля из файла, выдачу информации о студентах по фамилии, номеру группы, подсчет среднего												

	балла по выбранному предмету и по выбранной группе, а также упорядочения данных по фамилиям студентов или номеру группы.
23	Объявить структуру для хранения информации о результатах сдачи экзамена: номер по порядку, фамилия студента, номер зачетной книжки, оценка (неуд., уд., хор., отл.). Объявить массив «Экзаменационная ведомость» и поместить в него сведения о группе студентов. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение ведомости из файла, выдачу информации о студенте по первым буквам фамилии, подсчет процента студентов, сдавших экзамен на определенную оценку, а также упорядочение ведомости по номеру зачетки или фамилии.
24	Объявить структуру для хранения информации о спортсмене: фамилия, время в беге на 100 м., время в беге на 3 км., результаты прыжков в высоту и в длину. Объявить массив «Результаты соревнований» и поместить в него сведения о нескольких спортсменах. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение информации о результатах соревнований из файла, выдачу информации о спортсмене по какому-либо результату, определение лучшего спортсмена по введенному виду спорта, вывод всех данных, упорядоченных по любому из полей структуры.
25	Объявить структуру для хранения библиотечной карточки: раздел каталога, автор, наименование книги, количество страниц, год издания, издательство. Объявить массив карточек и поместить в него сведения о нескольких книгах. Разработать программу, позволяющую выполнить создание новой карточки, запись всех карточек в файл, чтение карточек из файла, выдачу информации о книге (книгах) по фамилии автора, названию или году издания; подсчет кол-ва книг в каждом разделе каталога; вывод упорядоченного массива карточек по разделам каталога или фамилиям авторов.
26	Объявить структуру для хранения информации о результатах сдачи экзамена: номер по порядку, фамилия студента, номер зачетной книжки, оценка (неуд., уд., хор., отл.). Объявить массив «Экзаменационная ведомость» и поместить в него сведения о группе студентов. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение ведомости из файла, выдачу информации о студенте по первым буквам фамилии, подсчет процента студентов, сдавших экзамен на определенную оценку, а также упорядочения ведомости по полю «Фамилия студента» в алфавитном порядке.
27	Объявить структуру для хранения библиотечной карточки: раздел каталога, автор, наименование книги, количество страниц, год издания, издательство. Объявить массив карточек и поместить в него сведения о нескольких книгах. Разработать программу, позволяющую выполнить создание новой карточки, запись всех карточек в файл, чтение карточек из файла, выдачу информации о книге (книгах) по фамилии автора, названию или году издания; подсчет кол-ва книг в каждом разделе каталога; вывод упорядоченного массива карточек по разделам каталога или фамилиям авторов.
28	Объявить структуру для хранения информации о результатах сдачи экзамена:

	номер по порядку, фамилия студента, номер зачетной книжки, оценка (неуд., уд., хор., отл.). Объявить массив «Экзаменационная ведомость» и поместить в него сведения о группе студентов. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение ведомости из файла, выдачу информации о студенте по первым буквам фамилии, подсчет процента студентов, сдавших экзамен на определенную оценку, а также подсчет среднего балла по группе.
29	Объявить структуру для хранения информации о результатах сдачи экзамена: номер группы, фамилия студента, название экзамена, оценка (неуд., уд., хор., отл.). Объявить массив «Табель успеваемости» и поместить в него сведения о студентах из разных групп. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение табеля из файла, выдачу информации о студентах по фамилии, номеру группы, подсчет среднего балла по выбранному студенту и по выбранной группе.
30	Объявить структуру для хранения информации о результатах сдачи экзамена: номер группы, фамилия студента, название экзамена, оценка (неуд., уд., хор., отл.). Объявить массив «Сессия» и поместить в него сведения о студентах из разных групп. Разработать программу, позволяющую выполнить создание новой записи, запись всех данных в файл, чтение табеля из файла, выдачу информации о студентах по фамилии, номеру группы, подсчет среднего балла по выбранному студенту и по выбранной группе.

8. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Каким образом объявляется структура в C++?
2. Каким образом объявляется объединение в C++? Приведите пример кода для создания объединения.
3. Каким образом объявляется перечисление в C++? Приведите пример кода для создания перечисления.
4. Создайте в программе объект на основе типа объединение (union). Докажите, что элементы объединения располагаются по одному адресу. Используйте указатели подходящих типов.
5. Выполните задание, используя объединения. Ввести с клавиатуры два произвольных символа. Составить число типа `short int`, байты которого совпадают с кодами введенных символов. Ввести с клавиатуры ещё два символа. Получить второе число типа `short int` аналогично первому. Распечатать *большее* из чисел. Числа и символы распечатать в шестнадцатеричной системе счисления.