**Structures**

**Structure is a composite object that includes elements of any data type.**

**Unlike an array, which is a homogeneous object, a structure can be heterogeneous.**.

**To create structure:**

**struct Name**

**{ elements or fields of structure;**

**};**

**For example:**

**struct student** // structure template specified or **new data type student** declared

**{char name[30];**

**int course;**

**char group[3];**

**int phone;**

**};**

To declare variables of type structure student, you can write:

**student stud1, stud2;**

Access to a specific element of the structure is carried out using an operation. **«.»**

First you need to access to the variable of the structure type, and then to its field

**For example:**

**cin>>stud1.name;**

**cout<<stud1.name;**

For the variable **stud2**, we use:

**сout<< stud2.group;**

If structure is already created, you can declare an array of its elements

**student stud\_ikm[200];**

**You can access to any element of an array of type structure through its index**

**stud\_ikm[0], stud\_ikm[l]** і т.і.

If we want to set the phone number of the 24th student (array element), then we first access to this element, and then to its field (phone):

**stud\_ikm[24].kurs**

Structure type variable can be **global, local** variable and **function argument**.

Arrays, structures, and arrays of structures can be used as structure elements.

For instance:

**struct address {**

**char city [30];**

**char street[30];**

**int house;**

**};**

**struct fulladdr {**

**address a;//** variable of the structure type **address**

**int room;**

**char name[30];**

**} f, g;** //f i g –variables of the structure type **fulladdr**

To access the field **house** for the variable **f** we use:

**f.a.house=101;**

**Example 1.** Create a program that implements the storage and processing of information about students using **data** structures. To store and process information about students of different courses from the one institute, create an array of data structures student, which contains 3 records.

**Student** structure includes the following fields:

-Surname of the student;

-data;

-course;

The "**data**" field is a separate structure with fields:

-address;

-telephone.

Organize the input of information about students in a separate function **input**.

In the main function main (), organize the printing of information about the students of the course, the number of which will be entered from the keyboard.

**Program:**

#include <iostream>

using namespace std;

struct data

{

char address[10];

int tel;

};

struct student

{

char surname[8];

data dst;

int course;

};

void input (student A[3]);//function for the filling information about students the title

int i;

int main()

{

student f1[3];//array of structure student

int k;

input(f1);

cout<<"Enter the course numder"<<endl;

cin>>k;

for (i=0;i<3;i++)

if (f1[i].course==k)

{cout<<"Surname: "<<f1[i].surname<<endl;

cout<<"Address: "<<f1[i].dst.address<<endl;}

return 0;

}

void input(student A[3]) //function for the filling information about students the title and body

{

for(i=0;i<3;i++)

{

cout<<"Surname "<<endl;

cin>>A[i].surname;

cout<<"course "<<endl;

cin>>A[i].course;

cout<<"address "<<endl;

cin>>A[i].dst.address;

cout<<"tel"<<endl;

cin>>A[i].dst.tel;

}

}