

Programming and Problem Solving through C Language O Level / A Level

Chapter - 8 : Structures and Unions

Initialization - Initializing Structure Members

- The members of individual structure variable are initializing one by one or in a single statement.
- The example to initialize a structure variable is

```
struct employee
{
    int emp_id;
    char name[20];
    int salary;
    char address[30];
    int dept_no;
    int age;
};

struct employee e1={1 , "Hemant", 12000, "3 vikas colony new delhi" , 10,35);

e1.emp_id=1;
strcpy(e1.name,"Hemant");
e1.salary=12000;
strcpy(e1.address,"3 vikas colony new delhi");
e1.dept_no=10;
e1.age=35;
```

Accessing Structure Members

- The structure members cannot be directly accessed in the expression.
- They are accessed by using the name of structure variable followed by a dot and then the name of member variable.
- The method used to access the structure variables are e1.emp_id, e1.name, e1.salary, e1.address, e1.dept_no, e1.age.
- The data within the structure is stored and printed by this method using **scanf** and **printf** statement in c program.

Alternate way for Structure Variable

We can define the structure variable while defining the structure.

```
struct employee
{
    int emp_id;
    char name[20];
    int salary;
    char address[30];
    int dept_no;
    int age;
} e1, e2;
```

Program for array of structure variable

```
struct student{
    char name[50];
    int roll;
    float marks;
};
int main(){
    struct student s[10];
    int i;
    printf("Enter information of students:\n");
    for(i=0;i<10;++i)
    {
        s[i].roll=i+1;
        printf("\nFor roll number %d\n",s[i].roll);
        printf("Enter name: ");
        scanf("%s",s[i].name);
        printf("Enter marks: ");
        scanf("%f",&s[i].marks);
        printf("\n");
    }
    printf("Displaying information of students:\n\n");
    for(i=0;i<10;++i)
    {
        printf("\nInformation for roll number %d:\n",i+1);
        printf("Name: ");
        puts(s[i].name);
        printf("Marks: %.1f",s[i].marks);
    }
    return 0;
}
```

Structure Assignment

- The value of one structure variable is assigned to another variable of same type using assignment statement.
- If the e1 and e2 are structure variables of type employee then the statement e1 = e2; assign value of structure variable e2 to e1.
- The value of each member of e2 is assigned to corresponding members of e1.
- Individual structure members can be used like other variables of the same type.
- Structure members are accessed using the structure member operator (.), also called the dot operator, between the structure name and the member name.
- One major advantage is that a programmer can copy information between structures of the same type with a simple equation statement.

Example

```
struct employee
{
    int emp_id;
    char name[20];
    int salary;
    char address[30];
    int dept_no;
    int age;
};

void main()
{
    struct employee e1={1 , "Hemant", 12000, "3 vikas colony new delhi" , 10,35);
    struct employee e2;

    e2=e1; // all the data of members of e1 copied to members of e2

    printf("%d\n",e2.emp_id);
    printf("%s\n",e2.name);
    printf("%s\n",e2.address);
    printf("%d\n",e2.dept_no);
    printf("%d\n",e2.age);
}
```

Structures within Structures

- C language allows a variable of structure type to be a member of some other structure type.
- The syntax to define the structure within structure is

```
struct<struct_name>
{
    <data_type><variable_name>;
    struct<struct_name>
        { <data_type><variable_name>;
          .....}<struct_variable>;
    <data_type><variable_name>;
};
```

Example of Structure within Structure

```
#include <stdio.h>

struct Employee
{
    char ename[20];
    int ssn;
    float salary;
    struct date
    {
        int date;
        int month;
        int year;
    }doj;
}emp = {"Pritesh",1000,1000.50,{22,6,1990}};

int main(int argc, char *argv[])
{
    printf("\nEmployee Name   : %s",emp.ename);
    printf("\nEmployee SSN    : %d",emp.ssn);
    printf("\nEmployee Salary : %f",emp.salary);
    printf("\nEmployee DOJ     : %d/%d/%d", \
        emp.doj.date,emp.doj.month,emp.doj.year);

    return 0;
}
```

Accessing Structures within Structures

- The data member of structure within structure is accessed by using two period (.) symbols.
- The syntax to access the structure within structure is
struct_var.nested_struct_var.struct_member;
- For Example:
e1.doj.day;
e1.doj.month;
e1.doj.year;