C programming

Trainer : Nisha Dingare

Email : nisha.dingare@sunbeaminfo.com

Structure :

- Structure is a user-defined data type.
- Structure stores logically related (similar or non-similar) elements in contiguous memory location.
- Structure members can be accessed using "." operator via struct variable.
- Structure members can be accessed using "->" operator via struct pointer.
- Size of struct = Sum of sizes of struct members.
- If struct variable initialized partially at its point of declaration, remaining elements are initialized to zero.

```
// struct data-type declaration (global or local)
struct emp {
    int empno;
    char ename[20];
    double sal;
};
// struct variable declaration
struct emp e1 = {11, "John", 20000.0};
// print struct members
printf("%d%s%lf", e1.empno, e1.ename, e1.sal);
```

Structure Declaration :

- Structure declarations are generally done before main i.e. global declaration.
- We can also do it in a function.
- struct student {
- int roll_no;
- char name[10];
- float avg;
- };

/*initializing structure variable at its declaration*/

- void main()
- {
- struct student s1={1,”Nisha”,80};
- printf(“size=%d”,sizeof(s1));
- printf(“roll=%,name=%s,avg=%f”,s1.roll, s1.name, s1.avg);
- }

Array of structure :

- C does not limit a programmer to storing simple data types inside an array.
- User defined structures too can be elements of an array.
- Example : struct student s[10];
- Data of 10 students can be stored with the above example.

Passing Structure to a Function by Value and by Reference :

- A structure can be passed to any function from main function or from any sub function.
- Structure definition will be available within the function only.
- Example

```
struct student
{ int rollno; int age; };
void display(student s);  // passing structure by value in function argument
//display(st); // calling function

void show(student *s); // passing structure by reference in function argument
//show(&st); // calling function
```

Structures and Pointers :

- Just like a variable, you can declare a pointer pointing to a structure and assign the beginning address of a structure to it.

- **Pointer to Structure**

```
void main()
{
        struct student s1={10,"Sujata",78.67};
        struct student *ptr = &s1;
        printf("size=%d",sizeof(ptr));
        printf("roll=%d,nm=%s,avg=%f",ptr→roll, ptr→name, ptr→avg);
}
```

Nested Structures :

- One can define a structure which in turn can contain another structure as one of its members.
- Example:

```
typedef struct
{int dd;  int mm;  int yy;  }DATE;
typedef struct
{int rollno;
int marks;
struct
        {
        char fname[10];
        char mname[10];
        char lname[10];
        }name;
DATE dob;
}STUDENT;
```

Thank You !!