Accessing Structure Members in C:

1. Array elements are accessed using the Subscript variable, Similarly Structure members are accessed using dot [.] operator.

2. (.) is called as “Structure member Operator”.

3. Use this Operator in between “Structure name” & “member name”

Example:

```c
#include<stdio.h>
#include<conio.h>

struct Vehicle
{
    int wheels;
    char vname[20];
    char color[10];
} v1 = {4,"Maruti 800","White"};

void main()
{
    printf("Vehicle No of Wheels : %d",v1.wheels);
    printf("Vehicle Name           : %s",v1.vname);
    printf("Vehicle Color          : %s",v1.color);
    getch();
}
```

Structure within Structure: Nested Structure

- Structure written inside another structure is called as nesting of two structures.
- Nested Structures are allowed in C Programming Language.
- We can write one Structure inside another structure as member of another structure.

I- Way of declaration of nested structure:

```c
struct date
{
    int date;
    int month;
    int year;
};
```
```c
struct Employee
{
    char ename[20];
    int ssn;
    float salary;
    struct date doj;
} emp1;

Way of Accessing Elements of Nested Structure :

1. Structure members are accessed using dot operator.
2. 'date' structure is nested within Employee Structure.
3. Members of the 'date' can be accessed using 'employee'
4. emp1 & doj are two structure names (Variables)

Explanation of Nested Structure :

Accessing Month Field: emp1.doj.month
Accessing day Field : emp1.doj.day
Accessing year Field: emp1.doj.year

II- Way of declaration of embedded structures

struct Employee
{
    char ename[20];
    int ssn;
    float salary;
    struct date
    {
        int date;
        int month;
        int year;
    } doj;
} emp1;

Accessing Nested Members :

Accessing Month Field : emp1.doj.month
Accessing day Field : emp1.doj.day
Accessing year Field : emp1.doj.year
```
Example:

```c
#include <stdio.h>
#include<conio.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}};

Void main()
{
    printf("Employee Name   : %s",emp.ename);
    printf("Employee SSN    : %d",emp.ssn);
    printf("Employee Salary : %f",emp.salary);
    printf("Employee DOJ    :
        %d/%d/%d",
        emp.doj.date,emp.doj.month,emp.doj.year);

    getch();
}
```

**Pointer to structure**: Pointer which stores address of structure is called as “**Pointer to Structure**“.

**Explanation**:
1. `sptr` is **pointer to structure** address.
2. `->` and `(*)` both represent the same.
3. These operators are used to access data member of structure by using **structure’s pointer**.

**Program**:

```c
#include<stdio.h>

struct team {
    char *name;
    int members;
    char captain[20];
}
t1 = {"India",11,"Dhoni"}, *sptr = &t1;

int main()
{

printf("\nTeam : %s",(*sptr).name);
printf("\nMembers : %d",sptr->members);
printf("\nCaptain : %s",(*sptr).captain);

return 0;
}

---

**Passing Structure to Function in C Programming**

1. Structure can be passed to **function as a Parameter**.
2. function can also Structure as **return type**.
3. Structure can be passed as follow

**Example:**

```
#include<stdio.h>
#include<conio.h>

struct Example
{
    int num1;
    int num2;
}s[3];

void accept(struct Example *sptr)
{
    printf("\nEnter num1 : ");
    scanf("%d",&sptr->num1);
    printf("\nEnter num2 : ");
    scanf("%d",&sptr->num2);
}

void print(struct Example *sptr)
{
    printf("\nNum1 : %d",sptr->num1);
    printf("\nNum2 : %d",sptr->num2);
}

void main()
{
    int i;
```
clrscr();
for(i=0;i<3;i++)
accept(&s[i]);

for(i=0;i<3;i++)
print(&s[i]);

getch();
}

Accessing Element in Structure Array

1. Array of Structure can be accessed using dot [.] operator.
2. Here Records of 3 Employee are Stored.
3. ‘for loop’ is used to Enter the Record of first Employee.
4. Similarly ‘for Loop’ is used to Display Record.

Example:

```c
#include<stdio.h>
#include<conio.h>

struct Employee
{
   int ssn;
   char ename[20];
   char dept[20];
}; emp[3];

//Enter the Employee Details
for(i=0;i<3;i++)
{
   printf("Enter the Employee Details : ");
   scanf("%d %s %s",&emp[i].ssn,emp[i].ename,emp[i].dept);
}

//Print Employee Details
for(i=0;i<3;i++)
{
   printf("Employee SSN : %d",emp[i].ssn);
}
```
printf("Employee Name : %s", emp[i].ename);
printf("Employee Dept : %d", emp[i].dept);
getch();