

ARRAYS OF STRUCTURES

Arrays of structures means that the structure variable would be an array of objects, each of which contains the member elements declared within the structure construct.

Why would need an array of structures

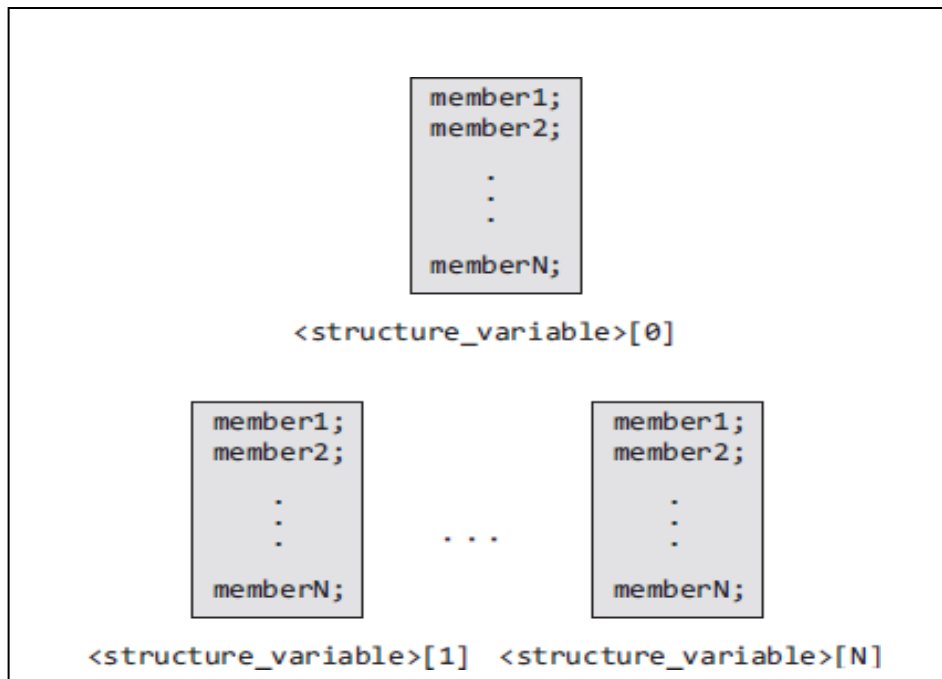
1. In a class, we do not have just one student. But there may be at least 60 students. So, the same definition of the structure can be used for all the 30 students. This would be possible when we make an array of structures.
2. Another example where an array of structures is desirable is in case of an organization. An organization has a number of employees. So, defining a separate structure for every employee is not a viable solution. So, here we can have a common structure definition for all the employees.

Now, to assign values to the ith we can write as

```
stud[i].r_no    =    09;
stud[i].name    =
"RASHI"; stud[i].course
= "MCA"; stud[i].fees =
60000;
```

In order to initialize the array of structure variables

```
struct student stud[3][4] = { {01, "Aman", "BCA", 45000}, {02,
"Aryan", "BCA", 60000}, {03, "John", "BCA", 45000} };
```



Write a program to print the tickets of the borders of a boat using array of structures with initialization in the program.

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

```
struct boat // declaration of structure //
```

```
{  
char name[20];  
int seatnum;  
float fare;  
};
```

```
int main()
```

```
{  
int i;
```

```
struct boat ticket[4][3]={{“Vikram”,1,15.50},{“Krishna”,2,15.50},  
{“Ramu”,3,25.50},{“Gouri”,4, 25.50 } };
```

```
printf(“\n passenger Ticket num. Fare”);
```

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

```
printf(“\n %s %d %f”, ticket[i].name,ticket[i].seatnum,ticket[i].fare);
```

```
return 0;
```

```
}
```

Output:

Passenger	Ticket num.	Fare
Vikram	1	15.500000
Krishna	2	15.500000
Ramu	3	25.500000
Gouri	4	25.500000

C program to generate salary slip of employees using structures

```
#include<stdio.h>
struct emp
{
    int empno ;
    char name[10];
    int bpay, allow, ded, npay ;
} e[10];

void main()
{
    int i, n ;

    printf("Enter the number of employees : ") ;
    scanf("%d", &n) ;

    for(i = 0 ; i < n ; i++)
    {
        printf("\nEnter the employee number : ") ;
        scanf("%d", &e[i].empno) ;

        printf("\nEnter the name : ") ;
        scanf("%s", e[i].name) ;

        printf("\nEnter the basic pay, allowances & deductions : ") ;scanf("%d
%d %d", &e[i].bpay, &e[i].allow, &e[i].ded) ;

        e[i].npay = e[i].bpay + e[i].allow - e[i].ded ;
    }
    printf("\nEmp. No.\t Name \t Salary \n") ;
    printf("\n *****") ;
    for(i = 0 ; i < n ; i++)
    {
        printf("%d \t %s \t %f \t ", e[i].empno,e[i].name, e[i].npay) ;
    }
    return 0;
```

Enter the employee number 2

Enter the employee number: 1001

Enter the employee number: Rina

Enter the basic pay, allowances & deductions: 75000 10000 2000

Enter the employee number: 2001

Enter the employee number: Bina

Enter the basic pay, allowances & deductions: 80000 10000 3000

Emp.No. NetSalar

www.binils.com

Write a program to read and display the information of all the students in a class. Then edit the details of the ith student and redisplay the entire information.

```
#include
<stdio.h>
#include
<string.h>
struct student
{
int roll_no;
char
name[80];
int fees;
char DOB[80];
};
int main()
{
struct student
stud[50]; int n, i,
num, new_rollno;
int new_fees;
char new_DOB[80],
new_name[80]; clrscr(); //
clear screen
printf("\n Enter the number of
students : ");scanf("%d", &n);
for(i=0;i<n;i++)
{
printf("\n Enter the roll
number : "); scanf("%d",
&stud[i].roll_no);

printf("\n Enter the
name : ");
gets(stud[i].name);

printf("\n Enter the
fees : ");
scanf("%d",&b>stud[i].
fees);

printf("\n Enter the
DOB : ");
gets(stud[i].DOB);
}

for(i=0;i<n;i++)
{
```

```
printf("\n *****DETAILS OF STUDENT
%d*****", i+1); printf("\n ROLL No. = %d",
stud[i].roll_no);
printf("\n NAME = %s",
stud[i].name); printf("\n FEES
= %d", stud[i].fees); printf("\n
DOB = %s", stud[i].DOB);
}
```

```
printf("\n Enter the student number whose record has to
be edited : ");scanf("%d", &num);
```

```
printf("\n Enter the new roll
number : "); scanf("%d",
&new_rolno);
```

```
printf("\n Enter the new
name : ");gets(new_name);
```

```
printf("\n Enter the new
fees : "); scanf("%d",
&new_fees);
```

www.binils.com

```
printf("\n Enter the new
DOB : ");gets(new_DOB);

stud[num].roll_no = new_rollno;
strcpy(stud[num].name, new_name);
stud[num].fees = new_fees;
strcpy (stud[num].DOB, new_DOB);

for(i=0;i<n;i++)
{
printf("\n *****DETAILS OF STUDENT
%d*****", i+1); printf("\n ROLL No. = %d",
stud[i].roll_no);
printf("\n NAME = %s",
stud[i].name); printf("\n FEES
= %d", stud[i].fees); printf("\n
DOB = %s", stud[i].DOB);
}
getch();
return 0;
}
```

Output

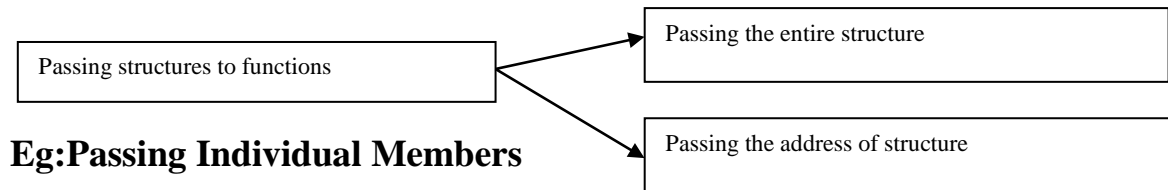
```
Enter the number of students : 2Enter
the roll number : 1 Enter the name : kirti
Enter the fees : 5678 Enter the DOB : 9-
9- 99
```

```
Enter the roll number : 2 Enter the
name : kangana Enter the fees : 5678
Enter the DOB : 27- 8- 99
*****DETAILS OF STUDENT
1*****ROLL No. = 1 NAME =
kirtiFEES = 5678
DOB = 9- 9- 99
*****DETAILS OF STUDENT
2*****ROLL No. = 2 NAME =
kanganaFEES = 5678
DOB = 27- 8 -99
```

```
Enter the student number whose record has to be edited : 2Enter the
new roll number : 2 Enter the new name : kangana khullarEnter the
new fees : 7000
Enter the new DOB : 27- 8 -92
```

```
*****DETAILS OF STUDENT
1*****ROLL No. = 1 NAME =
kirtiFEES = 5678
DOB = 9 -9 -99
*****DETAILS OF STUDENT 2*****
ROLL No. = 2
NAME = kangana
khullarFEES = 7000 DOB
```

STRUCTURES AND FUNCTIONS



Eg: Passing Individual Members

```
#include  
<stdio.h>  
typedef struct  
{  
    int  
    x;  
    int  
    y;  
}POINT;
```

```
void display(int, int);    // function declaration
```

```
int main()  
{  
    POINT p1 = {2, 3};  
    display(p1.x, p1.y); // function call  
    return 0;  
}
```

Output

The coordinates of the point are: 2 3

```
void display(int a, int b)    // function definition  
{  
    printf(" The coordinates of the point are: %d %d", a, b);  
}
```

Eg: Passing the Entire Structure

```
#include<stdio.h>  
typedef struct  
{  
    int x;  
    int y;  
}POINT;  
  
void display(POINT);  
int main()  
{  
    POINT p1 = {2, 3};  
    display(p1);  
    return 0;  
}  
void display(POINT p)  
{
```


www.binils.com for Anna University | Polytechnic and schools
`printf("The coordinates of the point are: %d %d", p.x, p.y);`
`}`

www.binils.com

- **Passing Structure by Value**
- **Passing Structure by Reference**

Passing Structure by Value

In this approach, the structure object is passed as function argument to the definition of function, here object is representing the members of structure with their values.

Program

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

```
struct Employee
{
    int Id;
    char
    Name[25];int
    Age;
    long Salary;
};
```

```
void Display(struct Employee);
```

```
void main()
```

```
{
    struct    Employee    Emp    =
```

```
{1,"Kumar",29,45000};Display(Emp);
```

```
}
```

```
void Display(struct Employee E)
```

```
{
```

```
    printf("\n\nEmployee    Id    :    %d",E.Id);
```

```
    printf("\nEmployee    Name    :    %s",E.Name);
```

```
    printf("\nEmployee    Age    :    %d",E.Age);
```

```
    printf("\nEmployee Salary : %ld",E.Salary);
```

```
}
```

Output

Employee Id : 1

Employee Name : Kumar

Employee Age: 29

Employee Salary: 45000

www.binils.com

Passing Structure by Reference

In this approach, the reference/address structure object is passed as function argument to the definition of function.

Program

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

```
struct
```

```
Employee
```

```
{
```

```
    int Id;
```

```
    char
```

```
    Name[25];int
```

```
    Age;
```

```
    long Salary;
```

```
};
```

```
void Display(struct
```

```
Employee*);void main()
```

```
{
```

```
    struct Employee Emp =
```

```
{1,"Kumar",29,45000};Display(&Emp);
```

```
}
```

```
void Display(struct Employee *E)
```

```
{
```

```
    printf("\n\nEmployee Id : %d",E->Id);
```

```
    printf("\nEmployee Name : %s",E->Name);
```

```
    printf("\nEmployee Age : %d",E->Age);
```

```
    printf("\nEmployee Salary : %ld",E->Salary)
```

```
}
```

Output :

Employee Id: EmployeeName:

Kumar EmployeeAge:29

EmployeeSalary : 45000

www.binils.com