CHAPTER 9

STRUCTURES

Arrays can be used to represent a group of data items that belong to the same type, such as float , int. However, if we want to represent a collection of data items of different type using a single name, then we can't use an array. For that purpose known as structure is used.

Structure syntax:

```
struct  tag_name
{
        data_type member1;
        data_type member2;
        data_type member3;
};
```

ex:

```
struct book_bank{
        char title[20];
        char author[20];
        int pages[15];
        float price;
        };
```

tag name

members of structure

Note:

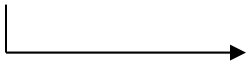
1) termination is with semicolon.
2) The tag_name can be used to declare structure variables of its type, later in program.

Formats

```
struct  book_bank
{
        char title[20];
        char author[20];
        int pages;
        float price;
} book1,book2,book3;
```

or

```
struct
{
        char title[20];
        char author[20];
        int pages;
        float price;
} book1,book2,book3;
```



structure variables

use of tag name is optional.

Giving values to members

book1.price = 12.00

where **.** operator is called **member of operator**.

Prog)wap to assign and retrive values from a structure.

Structure initialization

```
struct  std_rec{
                int weight;
                float height;
                };
```

struct std_rec std_rec1 = {60,180.75};

Comparison of structure variables

Person1 = person2;
Person1 ==person2;
Person != person2;

Array of structures:

Whenever the same structure is to be applied to a group of people, items, etc, in such situation , arrays of structure will be used.

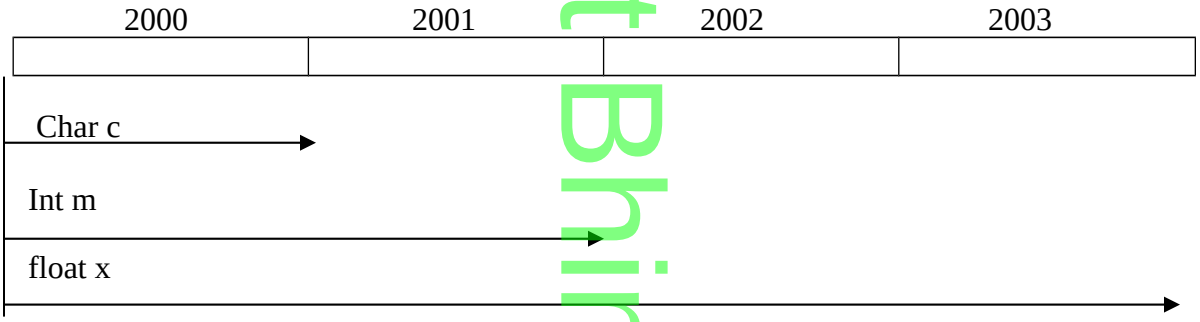
Ex:

Prog) to accepts the roll ni,name and marks obtained in 3 tests of three students of a class and display the roll no, name and marks of 3 tests and average.

```
Main()
{
        struct std_rec
        {
                int roll_no;
                char name[20];
                int m1;
                int m2;
                int m3;
                float avg;
        };

        int total;
        struct std_rec student[3];

        printf("type info of 3 students employee");
        for(I=0;I<3;I++)
        {
                printf("Enter The rollno of student %d\n",I+1);
                scanf("%d",&student[i].roll_no);
                printf("Enter name of student %d\n",I+1);
                scanf("%s",student[I].name);
                printf("Enter marks of student %d\n",I+1);
                scanf("%d %d %d",&student[I].m1,&student[I].m3,&student[I].m3);
        }

printf("\n-----------------------------------------------------------------------------------\n");
printf("Roll No        Name   Marks1        Marks2        Marks3        Average\n");
printf("\n-----------------------------------------------------------------------------------\n");
for(I=0i<3;I++)
{
        total = student[I].m1 + student[I].m2 + student[I].m3;
        student[I].avg = total /3.0;
        printf("%d      %s      %d      %d      %d      %5.2f\n",
        student[I].roll_no,student[I].name,student[I].m1,student[I].m3,student[I].m3,
        student[I].avg);
}
printf("\n-----------------------------------------------------------------------------------\n");

}/* EOM*/
```

UNION

```
Union item
{
        int m;
        float x;
        char c;
};
```



Storage of 4 bytes