STRUCTURE

structure is a user defined data type which is used to store the different datatype in single variable structure cover the drawback of array means array cannot store the different data type in single variable.

How to use the structure of programme

If we want to use the structure in programme for that c language provide the struct Keyword to us.

Syntax:- struct structure name

{

Datatype variablename;

Datatype variablename;

Datatype variablename;

};

e.g. struct student {

int id;

char name[80];

float marks;

};

If i declare the structure in programme but I want to use the structure member in programme

Then what can i do.

For that we need to create the variable of type structure or variable that structure.

What is role of structure variable in programme

If we declare then structure not allocates memory means structure member not stored in memory.but if we create the variable of that structure then structure member stored in

Memory.after that we can use it in programme.

How create the variable of structure .

Struct structurename variablename;

e.g. struct student st;// st is the variable of structe we can use the member of structure – through st.

or

another way of structure variable creation.

Struct structure name

{

Datatype variable name;

Datatype variable name;

Datatype variable name;

}

Variable1,variable2;

e.g. struct student

{

Int id;

Char name [80];

Float marks;

}

St,st1;

How to access the member of strucyure through the structure variable.

Syntax:- structvariable member of structure;

e.g. st id=100;

() dot is the member access operator which is used to access the structure member outside of structure anywhere in programme.

Memory representation given as per above example.

Id name marks

Integer value can store	String value can store herie	Float value can store here
100	102	182

Above program shows the memory area of st variable .

Because every character is 1 byte and its size is 80 and last variable allocate 4 byte memory.means overall size of st is 186 byte long.

Steps for structure use

1]declare the structure

2]create its variable

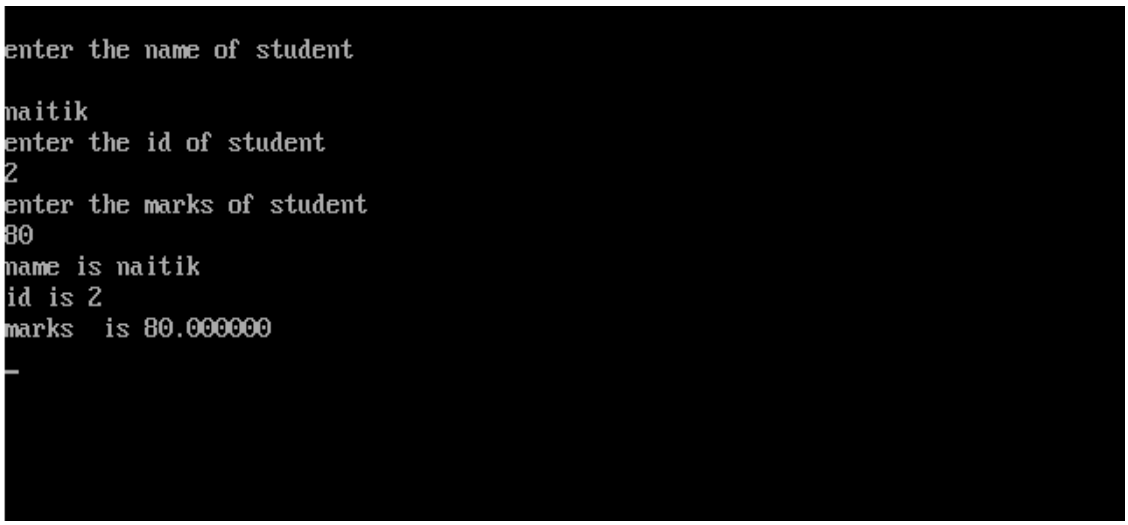
3]access its member using structure variable member of structure.

//Following program demonstrate the how to use the structure in programme.

```
#include<stdio.h>
#include<conio.h>
void main(){
struct student {
int id;
char name[80];
float marks;
};
struct student st;// create variable of type.
clrscr();
printf("enter the name of student \n");
scanf("%s",&st.name);//access name from structure
printf("enter the id of student \n");
scanf("%d",&st.id);//access id from structure
printf("enter the marks of student\n");
scanf("%f",&st.marks);
printf("name is %s \n",st.name);
```

```
printf("id is %d\n"st.id);

printf("marks  is %f\n",st.marks);

getch();

}
```

Output:



In above example we cannot store the more than one structure data in single variable if we required more than one student data then we need to create the variable of structure as per your requirement so your programme length increase unnecessary for that we can create the Array of structure.

ARRAY OF STRUCTURE

If we create the array of structure variable then we can store the more than one structure data in single variable.

How to create of array of the structure.

Syntax:- struct structurename variablename[size];

e.g. struct student st[5];

here the st is a variable of structure it will allocate the memory of five structure.

//Demonstrating the array of structure

```
#include<stdio.h>

#include<conio.h>

struct student   {   char name[20];   int roll,age;   };

main()   {   int i;   struct student S[10];   clrscr();

for(i=0;i<3;i++)     {  printf("\nEnter the details of the student : ");
```

```
scanf("%s %d %d",S[i].name,&S[i].roll,&S[i].age);     }   clrscr();

printf("\nThe details of the student are\n");   for(i=0;i<3;i++)

{     printf("\n%s \n%d \n%d\n\n",S[i].name,S[i].roll,S[i].age);

}  getch();   return 0;   }
```

output:-

What is the problem with above problem if we use the array then we cannot give the more data than specified aize and if we give the less element than size of array then remaining.

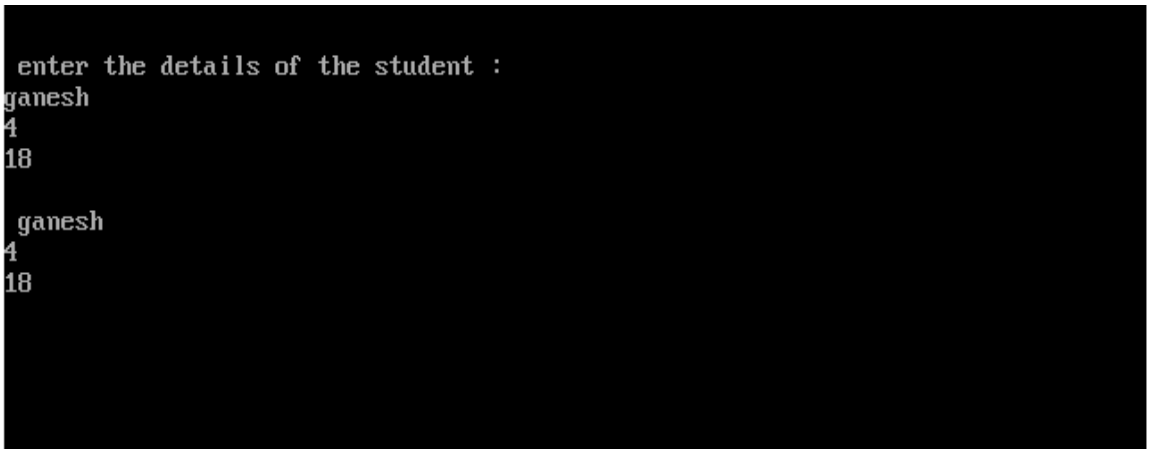
Memory will not use.

So we need to create the dynamic memory allocation with structure means we need to use the pointer with structure.

//Following programme demonstrate use of dynamic memory allocation with structure.

```
#include<stdio.h>

#include<conio.h>


Struct student {

char name[20];

int roll,age;

};

void main(){

Struct student *sptr;

clrscr();

sptr=(struct student *)malloc( sizeof (struct student));

printf("\n enter the details of the student :");

scanf("%S%d%d",&sptr->name,&sptr->roll,&sptr->age);

Printf("\n %s\n%d\n%d\n",sptr->name,sptr->roll,sptr->age);

getch();

}
```

Output:



How to pass the structure in function.

Syntax:- returntype functionname(struct structurename);

e.g. void accept(struct student);

if we want to pass the structure in function for that we need to declare structure at globally

//Following programme demonstrate function in structure

```
#include<stdio.h>
#include<conio.h>
struct student
{
int id,char,name[90];
float marks;
};
```

```
struct student st;
void main(){
void accept(struct student );
clrscr();
accept(st);//pass all structure member from here
getch();
}
void accept(struct student st){
printf("enter the name of student\n");
scanf("%s",&st.name);
printf("enter the id of student \n");
scanf("%d",&st.id);
printf("enter the marks of student \n");
pcanf("%f",&st.marks);
printf("name is %s ",&st.name);
printf("id is %d",&st.id);
printf("marks %f",&st.marks);
}
```

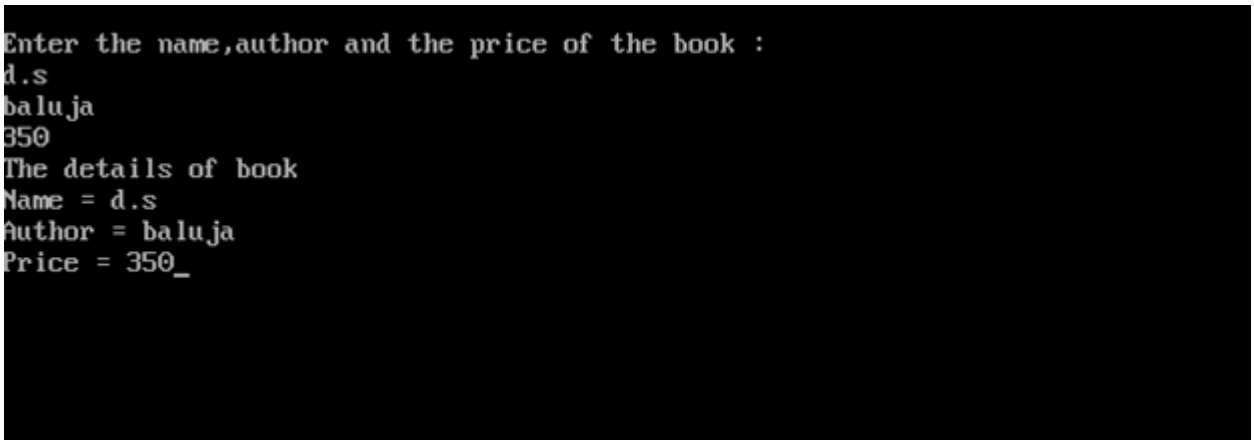
Output:-

SOLVED EXAMPLE ON STRUCTURE

//Demostrating the structure array variable.

```
#include<stdio.h>

#include<conio.h>

struct book   {   char name[20], author[20];   int price;   }B,B1,B2;

main()   {   struct book B;   clrscr();

printf("Enter the name,author and the price of the book : ");

scanf("%s %s %d",B.name,B.author,&B.price);   printf("The details of book");

printf("\nName = %s",B.name);   printf("\nAuthor = %s",B.author);

printf("\nPrice = %d",B.price);   getch();   return 0;   }
```

output:-



//Demonstrating the structure array

```
#include<stdio.h>

#include<conio.h>

struct customer  {  char name[20];  int age,acc_no;  int balance;  };

main()  {  struct customer C[10],C1;  int deposit,i;  clrscr();

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

printf("\nEnter the name,age,acc.no. and balance of customer  : ");

scanf("%s %d %d %d",C[i].name,&C[i].age,&C[i].acc_no,&C[i].balance);

printf("\nEnter the amount to be deposited : ");  scanf("%d",&deposit);

C[i].balance += deposit;  }

for(i=0;i<3;i++)  {  printf("\n\nthe details of customer %d",i+1);

printf("\nName = %s \nAge = %d \nAcc. No = %d \nBalance =
%d",C[i].name,C[i].age,C[i].acc_no,C[i].balance);

}  getch();  return 0;  }
```

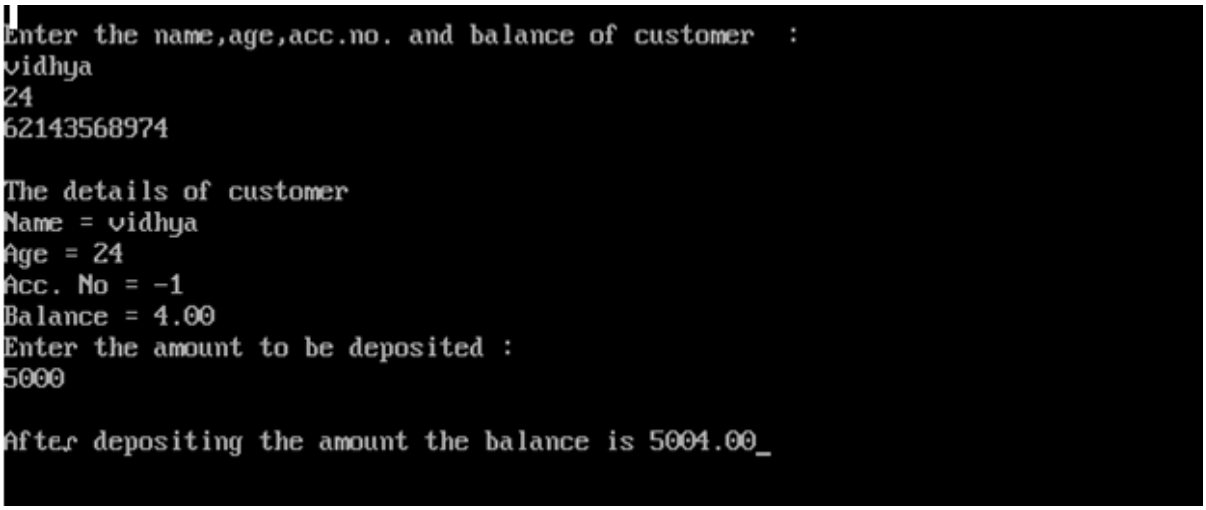
output:-

//Passing and rturning the structure variable from the function

```
#include<stdio.h>

#include<conio.h>

struct customer   {

char name[20];   int age,acc_no;   float balance;

};struct customer modify(struct customer);

main()   {    struct customer C;    clrscr();

printf("\nEnter the name,age,acc.no. and balance of customer  : ");
```

```
scanf("%s %d %d %f",C.name,&C.age,&C.acc_no,&C.balance);    printf("\nThe details of customer ");

printf("\nName = %s \nAge = %d \nAcc. No = %d \nBalance = %.2f",C.name,C.age,C.acc_no,C.balance);

C = modify(C);

printf("\nAfter depositing the amount the balance is %.2f",C.balance);    getch();    return 0;
}


struct customer modify(struct customer x)

{   int deposit;  printf("\nEnter the amount to be deposited : ");

scanf("%d",&deposit);   x.balance += deposit;   return x;
```

output:-




```
}
```

Demonstrating nesting of structure.

//Nesting of structures

```
#include<stdio.h>

#include<conio.h>

struct phone     {

int std;     long local;     };

struct address   {   int plotno;   char city[20];   struct phone P;   };

struct bdate   {   int date,year;   char month[10];   };

struct student   {   char name[20];   int roll,age;   struct address A;   struct bdate B;   };

main()   {   struct student S;   clrscr();   printf("\nEnter the name roll and age of the student :
");

scanf("%s %d %d",S.name,&S.roll,&S.age);   printf("\nEnter the plot number and the city : ");

scanf("%d %s",&S.A.plotno,S.A.city);   printf("\nEnter the date, month and the year : ");

scanf("%d %s %d",&S.B.date,S.B.month,&S.B.year);printf("\nEnter the std code and the local
number ");

scanf("%d %ld",&S.A.P.std,&S.A.P.local);   printf("\nThe details of the studemt are \n ");

printf("\n%s \n%d \n%d\n%d\n%s",S.name,S.roll,S.age,S.A.plotno,S.A.city);

printf("\nDate = %d Month = %s Year = %d",S.B.date,S.B.month,S.B.year);

printf("\nThe std code and the local number are  ");   printf("%d - %ld",S.A.P.std,S.A.P.local);

getch();   return 0;   }
```

output:-

```
Enter the name roll and age of the student :
soham
24
25

Enter the plot number and the city :
12
aurangabad

Enter the date, month and the year :
1   7      2015

Enter the std code and the local number
0240 6234444

The details of the studemt are

soham
24
25
12
aurangabad
Date = 1 Month = 7 Year = 2015
The std code and the local number are  240 - 6234444_
```

//Demo of pointer variable of structure

```
#include<stdio.h>

#include<conio.h>

struct student  {

char name[20];  int roll,age;  }*sptr;

main()  {  struct student S;  sptr = &S;  clrscr();  printf("\nEnter the details of the studemt
");  scanf("%s %d %d",sptr->name,&sptr->roll,&sptr->age);  printf("\nthe details of the
studemt are \n ");

printf("\n%s \n%d \n%d\n",sptr->name,sptr->roll,sptr->age);  getch();  return 0;  }
```

output:-

```
Enter the details of the studemt
rohan
1
20

the details of the studemt are

rohan
1
20
```

UNION

Union is also one kind of data type like structure.an union is variable that may hold (at different times)objects of different types and size,with the compiler keeping track of size.

And alignment requirements.union provides a way to manipulate different kind of data ina single area of storage.an union will contain one of the may different types of value (as long as only one is stored at a time).

The declaration and the usage of union is same as structure union hold only one value for one datatype (if we new assignment is made the previous value is automatically erased).

Declaring an union is similar to declaring structure. The genral form of union is.

Syntax:- union unionname {

Type number name 1;

Type number name2;

Type number name;

};

Union union name variable name;

//Following programme demonstrate the use of union.

```
#include<stdio.h>
#include<conio.h>
void main(){
union student {
int id;
char name[80];float marks;
}
union student st;//create variable at type.
printf("enter the name of student \n");
scanf("%s",&st.name);//access name from structure.
printf("enter the id of student\n");
scanf("%d",&st.id);
printf("enter the marks of student\n");
scanf("%f",&st.marks);
printf("name is %s \n",st.name);
printf("id is %d\n",st.id);
printf("marks is %d\n ",st.marks);
}

getch();
```

}

Output:-

Name is anil

Id is 2

Marks is 90