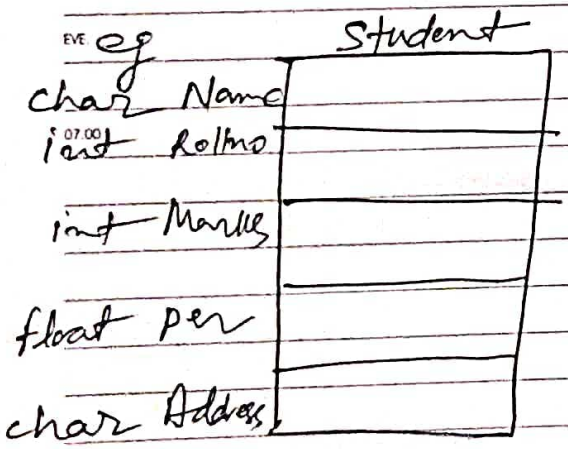
Structure

A structure is a collection or group of logically related variable with different datatypes (primitive or Derived) referenced underone name

⇒ The keyword of structure is struct.
By default datamembers of structure is public.

⇒ structure is a user defined datatype
It's datamember are accessed using . dot operator.

eg

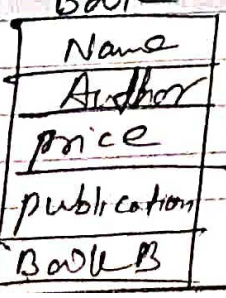



```
struct student
{
  char Name[30];
  int Rollno;
  int marks;
  float per;
  char Address[45];
};
student R;
```

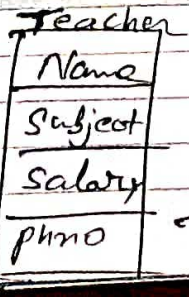
Accessed
R.Name;
R.Rollno;
R.Marks;
R.per;
R.Address;

NOTES :

Same



APPOINTMENTS :



PHONE/E-MAIL :

Syntax

```
struct structurename
{
    datatype variablename[size];
    datatype variablename[size];
    .
    .
    .
};
structurename object;
```


eg

```
struct teacher
{
char Name[30];
char subject[35];
float salary;
long int phoneno;
};
Teacher Z;
```

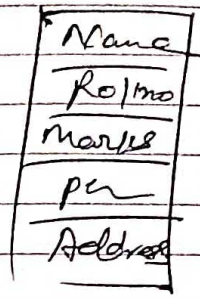
```
struct Book
{
char Name[40];
char Author[30];
float price;
char publication[45];
};
Book B;
```

NOTES :

APPOINTMENTS :

PHONE/E-MAIL :

Q Write a program in c++ to accept and display the details of student using structure.



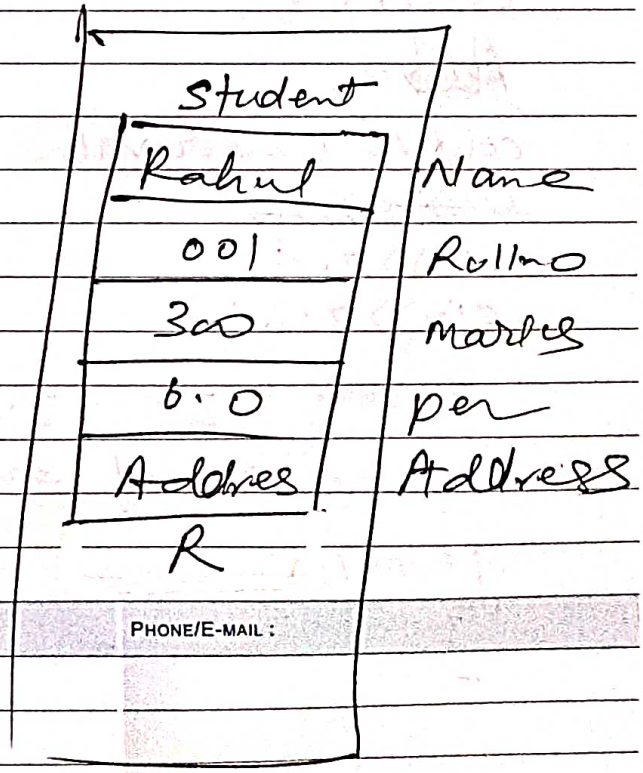

```
#include<iostream.h>
#include<conio.h>
void
struct student
{
  char Name[30];
  int Rollno;
  int marks;
  float per;
  char Address[45];
};
void main()
{
clrscr();
student R;
cout<<" Enter details of student";
cout<<"\n Enter Name";
cin>>R.Name;
cout<<"\n EnterRollno";
cin>>R.Rollno;
cout<<"\n Enter marks";
cin>>R.Marks
cout<<"\n Enter per";
cin>>R.per;
cout<<" Enter Address";
cin>>R.Address;
```

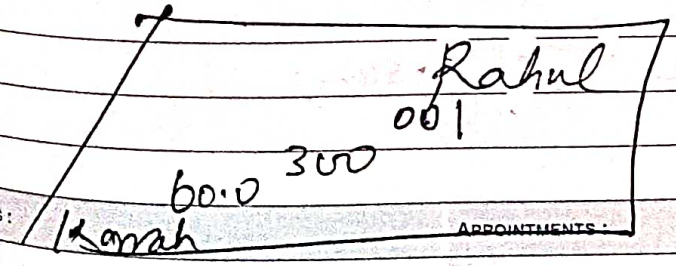

NOTES : APPOINTMENTS : PHONE/E-MAIL :

```
cout << " Display details of student";
cout << "\n Name " << R.Name;
cout << "\n Rollno " << R.Rollno;
cout << "\n marks " << R.Marks;
cout << "\n per " << R.per;
cout << "\n Address " << R.Address;
getch();
}
```

Enter details of student
Enter Name
Enter Rollno
Enter Marks
Enter per
Enter Address

Display details of student
Name Rahul
Rollno 001
Marks 300
per 60.0
Address Karrah





NOTES : APPOINTMENTS : PHONE/E-MAIL :

MARCH

21

12th Week • 080-285

Wednesday

Sum = a + b

2018

February 2018

S	M	T	W	T	F	S
				01	02	03
04	05	06	07	08	09	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28			

March 2018

S	M	T	W	T	F	S
				01	02	03
04	05	06	07	08	09	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Q Write a program in c++ to Add two number using structure.

09.00

10.00

11.00

12.00

LUNCH

03.00

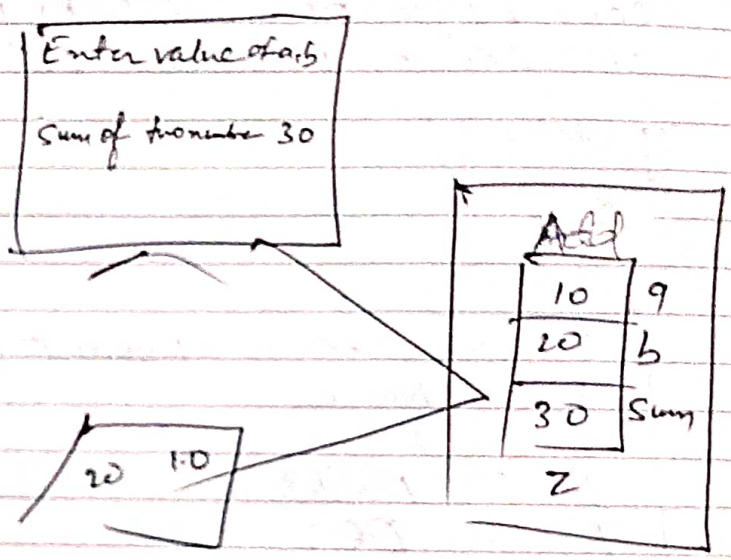
04.00

05.00

FYI

07.00

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

struct Add
{
int a;
int b;
int sum;
};

void main()
{
clrscr();
Add z;
cout<<"Enter value of a,b";

cin>>z.a;
cin>>z.b;

z.sum = z.a + z.b;
cout<<"sum of two number "<< z.sum;

getch();
}
```



NOTES :

APPOINTMENTS :

PHONE/E-MAIL :

MARCH
23
12th Week • 082-283
Friday

Nested Structure

Nested structure means structure within structure.
A structure containing object of another structure.

e.g

```
struct address
{
  int houseno;
  char city[30];
  char area[30];
  int pincode;
};

struct dateofbirth
{
   int dd;
   int mm;
   int yy;
};

struct student
{
  int rollno;
  char name[30];
  int marks;
  float per;
  address add;
  dateofbirth dob;
};
```

Syntax

```
struct structurename1
{
  datamember;
  datamember;
  :
};

struct structurename2
{
  datamember;
  datamember;
  :
};

struct structurename
{
  datamember;
  structurename1 obj1;
  structurename2 obj2;
  datamember;
};

structurename obj;
```

Student Z;

NOTES :
APPOINTMENTS
PHONE/E-MAIL :

APRIL

04

14th Week • 094-271

Wednesday

2018

March 2018

S	M	T	W	T	F	S
				01	02	03
04	05	06	07	08	09	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

April 2018

S	M	T	W	T	F	S
01	02	03	04	05	06	07
08	09	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Structure initialization

eg

```
struct Teacher
{
  char name[30];
  char subject[40];
  float salary;
  int age;
  long int phone;
};
Teacher T1;
```

T1 = {"Rahul", "computer", 10000.0, 28, {9905103748}};

Teacher T2 = {"Pappu", "physics", 20000, 35, 9977998887};

Teacher T3;

T3 = T1;

NOTES : APPOINTMENTS : PHONE/E-MAIL :

Extra

Initialization for basic program

Student z = {1, "Rahul", 300, 60.0, {5, "Feb", "Mathur", 825301}, {07, 01, 100}};