1. **Create a structure named company which has name, address, phone and noOfEmployee as member variables. Read name of company, its address, phone and noOfEmployee. Finally display these members' value.**

```
 #include <stdio.h>
#include <stdlib.h>
struct company
{
    char name[20],address[50];
    int phone,noOfEmployee;
};

int main(){

    struct company C1;
    printf("enter the details of the company");
    scanf("%s%s%d%d",C1.name,C1.address,&C1.phone,&C1.noOfEmployee);
    printf("Details of the company");
    printf("company name=%s \n company address=%s \n company phone=%d \n company noOfEmployee=%d ",C1.name,C1.address,C1.phone,C1.noOfEmployee);
}

```

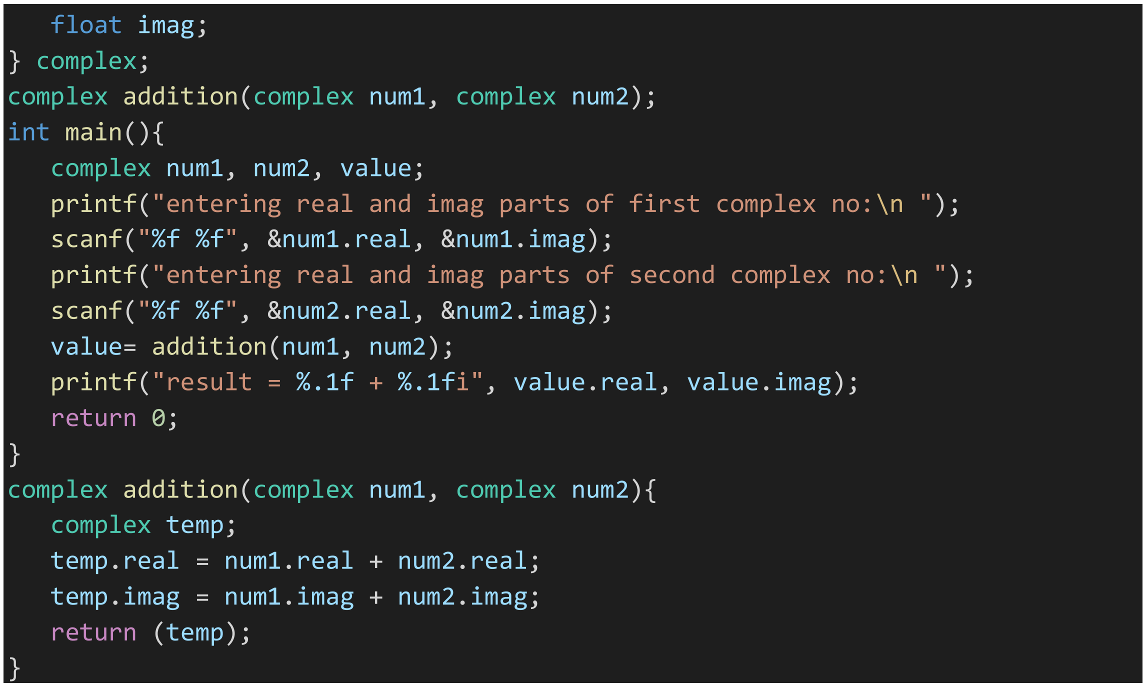
2. Write a program to enter to Cartesian coordinate points and display the distance between them.

```
 #include <stdio.h>
#include<math.h>
//function to find distance bewteen 2 points
void two_dis(float x1, float y1, float x2, float y2) {
   float dis = sqrt(pow(x2 - x1, 2) + pow(y2 - y1, 2) * 1.0);
   printf("Distance between 2 points are : %f", dis);
   return;
}
int main() {
   float x1 = 4;
   float y1 = 9;
   float x2 = 5;
   float y2 = 10;
   two_dis(x1, y1, x2, y2);
   return 0;
}
```

3. Write a function which accepts structure as argument and returns structure to the calling program.

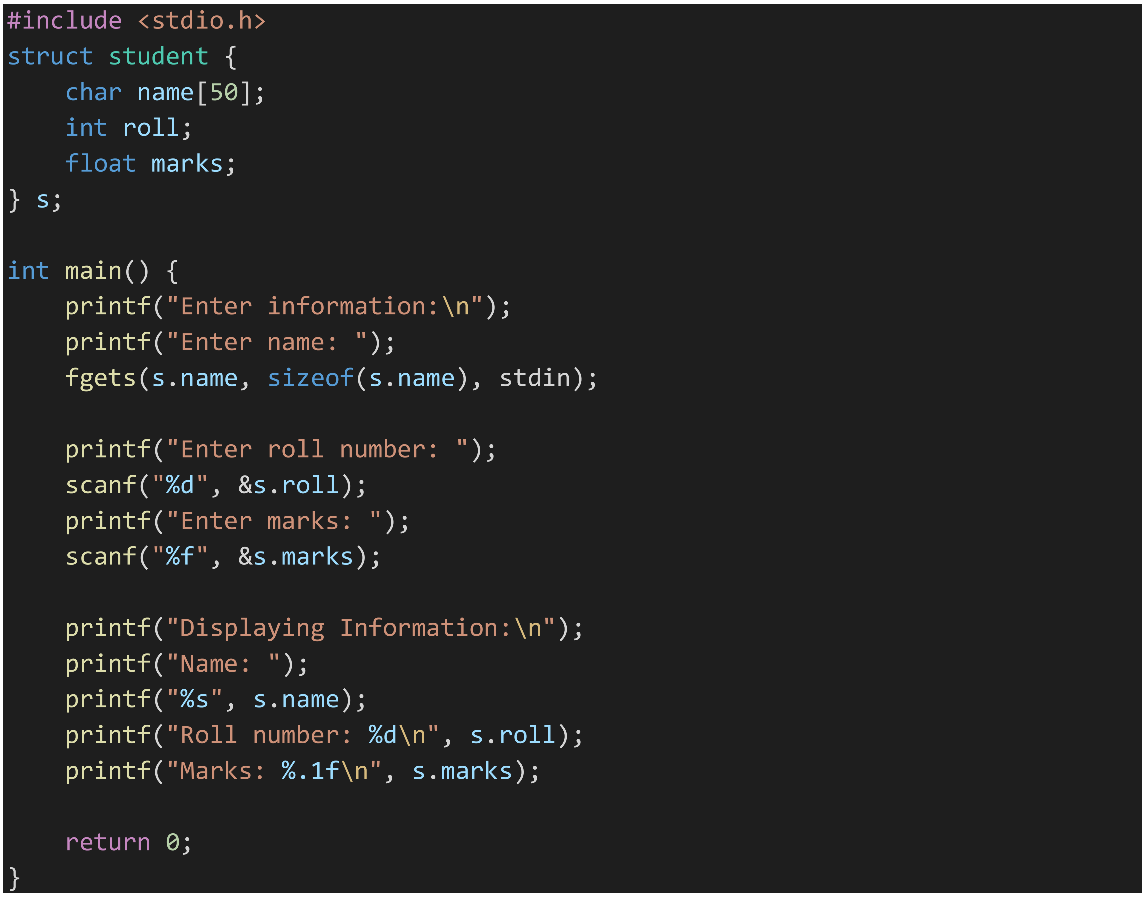
5. Define a structure "complex" (typedef) to read two complex numbers and perform addition, subtraction of these two complex numbers and display the result.

```
#include <stdio.h>
typedef struct complex{
   float real;
```



```
   float imag;
} complex;
complex addition(complex num1, complex num2);
int main(){
   complex num1, num2, value;
   printf("entering real and imag parts of first complex no:\n ");
   scanf("%f %f", &num1.real, &num1.imag);
   printf("entering real and imag parts of second complex no:\n ");
   scanf("%f %f", &num2.real, &num2.imag);
   value= addition(num1, num2);
   printf("result = %.1f + %.1fi", value.real, value.imag);
   return 0;
}
complex addition(complex num1, complex num2){
   complex temp;
   temp.real = num1.real + num2.real;
   temp.imag = num1.imag + num2.imag;
   return (temp);
}
```

6. Write a program to read RollNo, Name, Address, Age & average-marks of 12 students in the BCT class and display the details from function.



```
#include <stdio.h>
struct student {
    char name[50];
    int roll;
    float marks;
} s;

int main() {
    printf("Enter information:\n");
    printf("Enter name: ");
    fgets(s.name, sizeof(s.name), stdin);

    printf("Enter roll number: ");
    scanf("%d", &s.roll);
    printf("Enter marks: ");
    scanf("%f", &s.marks);

    printf("Displaying Information:\n");
    printf("Name: ");
    printf("%s", s.name);
    printf("Roll number: %d\n", s.roll);
    printf("Marks: %.1f\n", s.marks);

    return 0;
}
```

In this program, a structure `student` is created. The structure has three members: `name` (string), `roll` (integer) and `marks` (float).
Then, a structure variable `s` is created to store information and display it on the screen.

7. Write a program to show programming examples with union and enumerations

Union

```
#include <stdio.h>
union unionJob
{
   //defining a union
   char name[32];
   float salary;
   int workerNo;
} uJob;

struct structJob
{
   char name[32];
   float salary;
   int workerNo;
} sJob;

int main()
{
   printf("size of union = %d bytes", sizeof(uJob));
   printf("\nsize of structure = %d bytes", sizeof(sJob));
   return 0;
}
```

enumerations

```
#include <stdio.h>

enum designFlags {
      BOLD = 1,
      ITALICS = 2,
      UNDERLINE = 4
};

int main() {
      int myDesign = BOLD | UNDERLINE;
      //    00000001
      //  | 00000100
      //  ___________
      //    00000101

    printf("%d", myDesign);

    return 0;
}
```