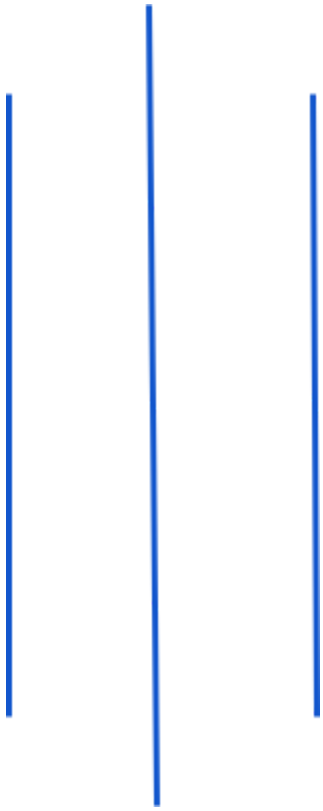
Tribhuvan University

Institute of Engineering

Thapathali Campus,Thapathali

LAB SHEET #5



Submitted by:

Name: Kishan Adhikari

Roll No. :THA077BCT021

Submitted to:

Department of Electronics and Computer Engineering

Date : 14th August 2021

Title:
Write a program to read RollNo, Name, Address, Age & marks in physics, C, math in 1 st semester of three students in your class and display the student details with average marks achieved.

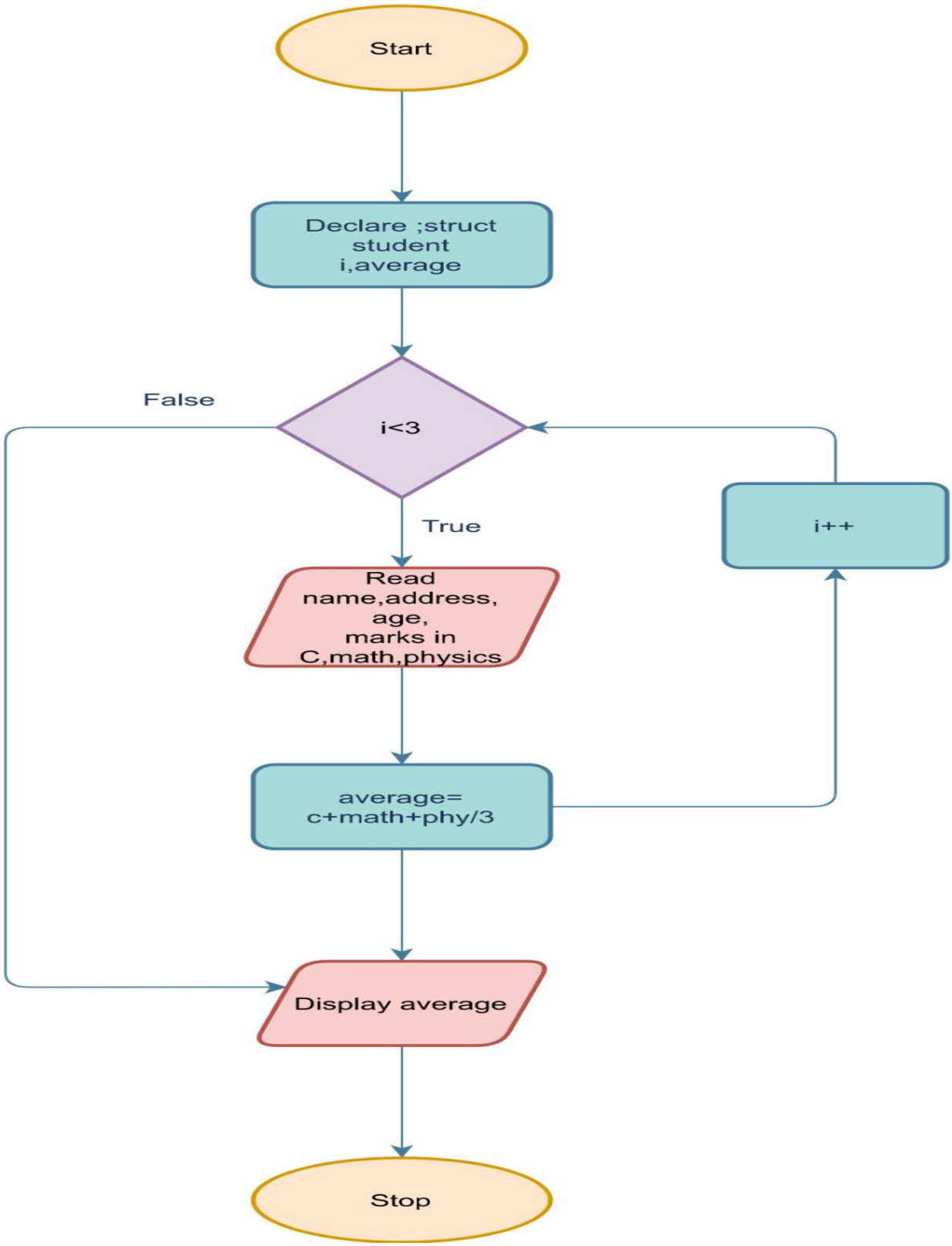
Problem Analysis:
The problem is to read Roll No,Name,Address,Age and marks of three student in physics,C and math in the first semester and display student detail with average marks. To do this we create array of structure to store above information of rollno(int), name(char[40]),Address(char[40),age(int) , marks(int),average for three subject under nested structure. Then we store information from keyboard and display average marks to console using printf function.
Calculation of average is done using formula as:
average=sum/total

Input Variables	Calculation	Output variables	Necessary Header file
Student (array of struct) rollno(int) name(char[40]) Address(char[40), age(int) , marks(int), average(float)	average=total_ marks/no_of_su bject	average(float)	stdio.h

Algorithm:
1. Start
2. Declare array of structure as student :
3. Declare variable inside structure as : name,roll,address,struct marks : c,physics,math
4. Declare average(float) variable

5. Store information of three student
6. Calculate average as :total/3
7. Display average of three student
8. Stop

Flowchart:

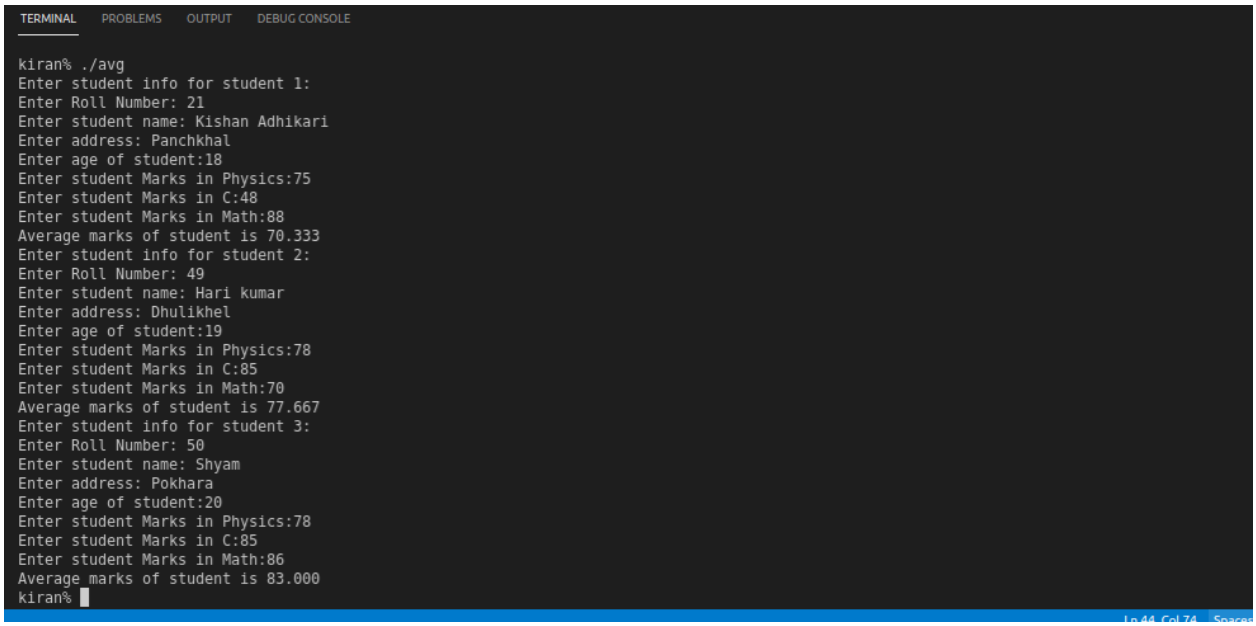

Start
Declare ;struct student i,average
i<3
False
True
Read name,address, age, marks in C,math,physics
average= c+math+phy/3
i++
Display average
Stop

Source code:

```
/* @Author: Kishan Adhikari
  @filename:avg.c
  @Created Date: 2078/05/08
  @Description:program to read RollNo, Name, Address, Age & marks in physics, C, math in 1 st semester of three students in your class and display the student details with average marks achieved.
*/

#include <stdio.h>
struct student
{
 int RollNo;
 char Name[40];
 char Address[40];
 int age;
 struct subject
 {
   int marks;
 } physics, C, Math;

} info;

int main()
{
 float average;
 for (int i = 0; i < 3; i++)
 {
   printf("Enter student info for student %d:\n", i + 1);
   printf("Enter Roll Number: ");
   scanf("%d", &info.RollNo);
   printf("Enter student name: ");
   scanf(" %[^\n]s", info.Name);
   fflush(stdin);
   printf("Enter address: ");
   scanf(" %[^\n]s", info.Address);
   fflush(stdin);
   printf("Enter age of student:");
   scanf("%d", &info.age);
   printf("Enter student Marks in Physics:");
```

```
    scanf("%d", &info.physics.marks);
    printf("Enter student Marks in C:");
    scanf("%d", &info.C.marks);
    printf("Enter student Marks in Math:");
    scanf("%d", &info.Math.marks);
    average = (info.physics.marks + info.C.marks + info.Math.marks) / 3.0;
    printf("Average marks of student is %.3f\n", average);
 }
 return 0;
}
```

Output:



Discussion and Conclusion:

From this lab , I understood the basics of structure,array of structure,and using them to store information about different things.

Title:

Write a program to read RollNo, Name, Address, Age & marks in physics, C, math in 1stsemester of three students in your class. Store the records into a file std.txt located at D:\drive. Display the student details with average marks achieved (use data files record I/O).

Source Code:

```
/*
@Author:Kishan Adhikari
@Filename:avgfile.c

@Description:Write a program to read RollNo, Name, Address, Age & marks in physics, C, math in 1stsemester of three students in your class. Store the records into a file std.txt located at
```

```
D:\drive. Display the student details with average marks achieved (use
data files record I/O).
*/
#include <stdio.h>
#include <stdlib.h>
float average;
struct student
{
 int RollNo;
 char Name[40];
 char Address[40];
 int age;
 struct subject
 {
   int marks;
 } physics, C, Math;
};
float avg[3];

void display(struct student[12]);
int main()
{
 int i;
 struct student s[12];

 FILE *fptr;
 fptr = fopen("std.txt", "w+");
 if (fptr == NULL)
 {
   printf("Error opening file\n");
   exit(0);
 }

 for (i = 0; i < 3; i++)
 {
   printf("Enter student info for student %d:\n", i + 1);

   printf("Enter Roll Number: ");
   scanf("%d", &s[i].RollNo);
   printf("Enter student name: ");
```

```
    scanf(" %[^\n]s", s[i].Name);
    fflush(stdin);
    printf("Enter address: ");
    scanf(" %[^\n]s", s[i].Address);
    fflush(stdin);
    printf("Enter age of student:");
    scanf("%d", &s[i].age);
    printf("Enter student Marks in Physics:");
    scanf("%d", &s[i].physics.marks);
    printf("Enter student Marks in C:");
    scanf("%d", &s[i].C.marks);
    printf("Enter student Marks in Math:");
    scanf("%d", &s[i].Math.marks);
    avg[i] = (s[i].physics.marks + s[i].C.marks + s[i].Math.marks) / 3;
  }
  fwrite(s, sizeof(struct student), 3, fptr);
  printf("Successfully saved\n");
  fclose(fptr);
  printf("****Displaying the content of file*****\n");
  FILE *file_ptr;
  file_ptr = fopen("std.txt", "r");
  fread(s, sizeof(s), 12, file_ptr);
  for (i = 0; i < 12; i++)
  {
    printf("Name: %s\n", s[i].Name);
    printf("Roll no: %d\n", s[i].RollNo);
    printf("Address : %s", s[i].Address);
            printf("Physics:    %d\tC:    %d\t    Math:%d\t\Average:%.2f\n",
s[i].physics.marks, s[i].C.marks, s[i].Math.marks, avg[i]);
  }
  fclose(file_ptr);
  return 0;
}
```

Output:

```
TERMINAL    PROBLEMS    OUTPUT    DEBUG CONSOLE

kiran% ./avg
Enter student info for student 1:
Enter Roll Number: 21
Enter student name: Kishan Adhikari
Enter address: Panchkhal
Enter age of student:18
Enter student Marks in Physics:45
Enter student Marks in C:32
Enter student Marks in Math:86
Enter student info for student 2:
Enter Roll Number: 49
Enter student name: Samir Timalsina
Enter address: Dhulikhel
Enter age of student:19
Enter student Marks in Physics:78
Enter student Marks in C:85
Enter student Marks in Math:88
Enter student info for student 3:
Enter Roll Number: 50
Enter student name: Rick Sanchez
Enter address: California
Enter age of student:62
Enter student Marks in Physics:99
Enter student Marks in C:98
Enter student Marks in Math:99
Successfully saved
****Displaying the content of file*****
Name: Kishan Adhikari
Roll no: 21
Address : PanchkhalPhysics: 45  C: 32    Math:86        Average:54.00
Name: Samir Timalsina
Roll no: 49
Address : DhulikhelPhysics: 78  C: 85    Math:88        Average:83.00
Name: Rick Sanchez
Roll no: 50
Address : CaliforniaPhysics: 99 C: 98    Math:99        Average:98.00
```

Discussion and Conclusion:

From this lab , I understood the basics of file handling, Structure as well as solving problems using C language.

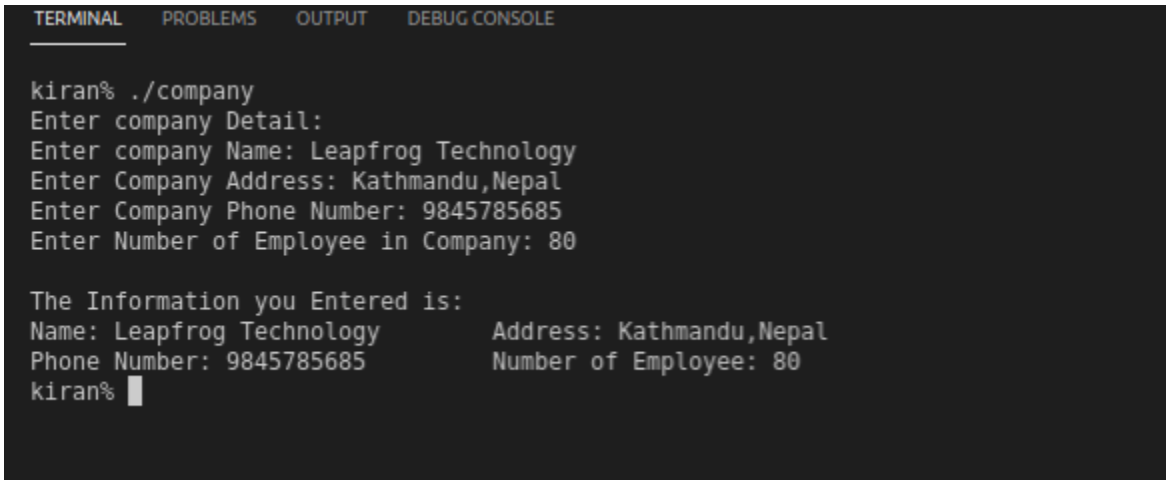
Title:

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.

Source Code:

```
/* @Author: Kishan Adhikari
 @Filename: company.c
 @Created Date: 2078/05/10
 @Description: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>
struct company
{
 char name[100];
 char address[100];
 long phone;
 int no_of_Employee;
} comp;
int main()
{
 printf("Enter company Detail:\n");
 printf("Enter company Name: ");
 scanf(" %[^\n]s", comp.name);
 printf("Enter Company Address: ");
 scanf(" %[^\n]s", comp.address);
 printf("Enter Company Phone Number: ");
 scanf("%ld", &comp.phone);
 printf("Enter Number of Employee in Company: ");
 scanf("%d", &comp.no_of_Employee);
 printf("\nThe Information you Entered is:\n");
 printf("Name: %s\t Address: %s\nPhone Number: %08ld\t Number of Employee: %d\n", comp.name, comp.address, comp.phone, comp.no_of_Employee);
 return 0;
}
```

Output:



Title:

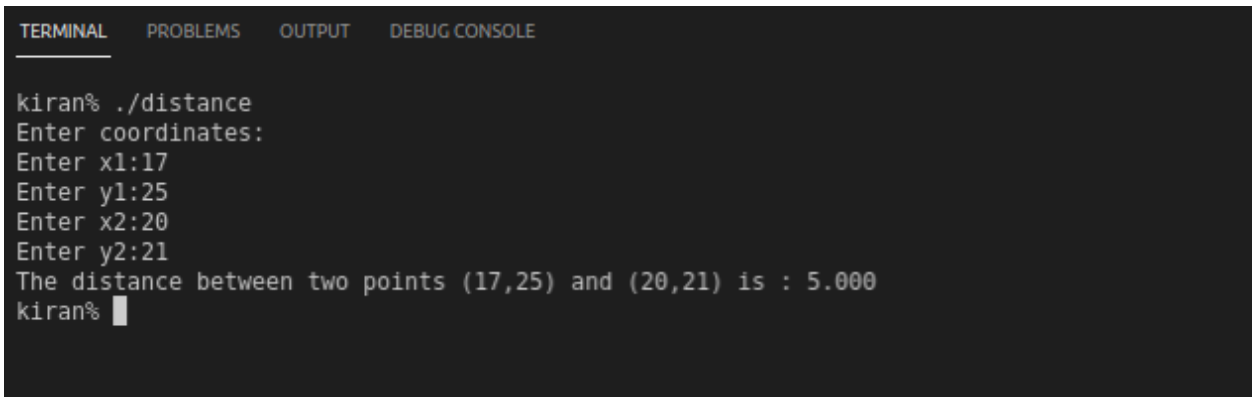
Write a program to enter Cartesian coordinate points (using structure) and display the distance between them.

Source Code:

```
/* @Author: Kishan Adhikari
 @Filename: distance.c
 @Created Date: 2078/05/10
  @Description:program to enter to Cartesian coordinate points (using structure) and display the distance between them
*/
#include <stdio.h>
#include <math.h>

struct coordinate
{
 long x_coordinate;
 long y_coordinate;
} a, b;
```

```
float calculate_distance()
{
 float distance;
  distance = sqrt(((b.x_coordinate - a.x_coordinate) * (b.x_coordinate - a.x_coordinate)) + ((b.y_coordinate - a.y_coordinate) * (b.y_coordinate - a.y_coordinate)));
 if (distance < 0)
 {
   distance = (-1) * distance;
 }
 return distance;
}
void main()
{
 printf("Enter coordinates:\n");
 printf("Enter x1:");
 scanf("%ld", &a.x_coordinate);
 printf("Enter y1:");
 scanf("%ld", &a.y_coordinate);
 printf("Enter x2:");
 scanf("%ld", &b.x_coordinate);
 printf("Enter y2:");
 scanf("%ld", &b.y_coordinate);
  printf("The distance between two points (%ld,%ld) and (%ld,%ld) is : %.3f\n", a.x_coordinate, a.y_coordinate, b.x_coordinate, b.y_coordinate, calculate_distance());
}
```

Output:



Title:

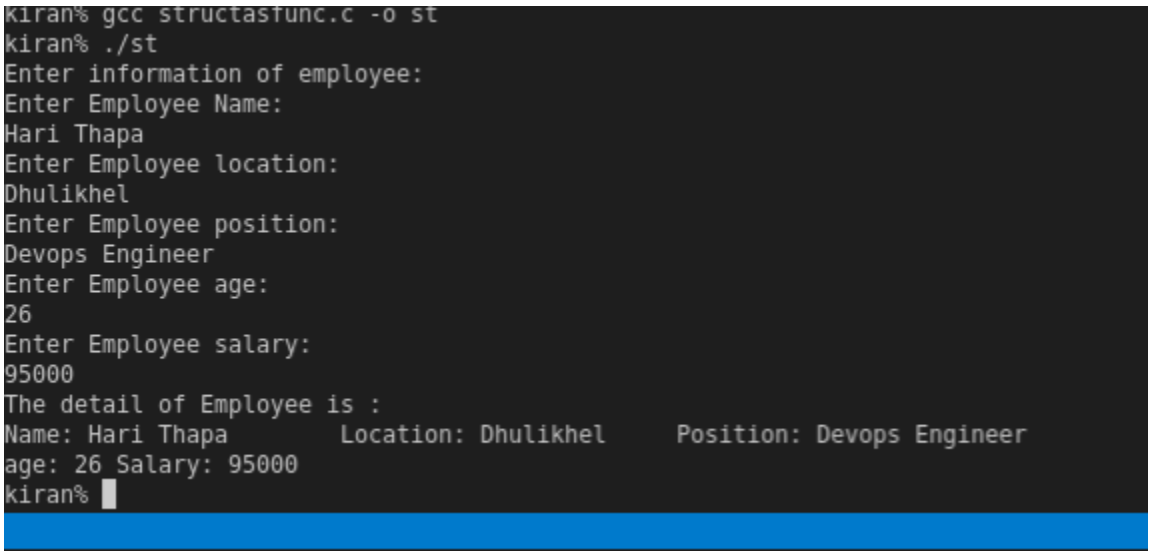
Write a function which accepts structure as argument and returns structure to the calling program.

Source Code:

```
/* @Author: Kishan Adhikari
 @Filename: structandfunc.c
 @Created Date: 2078/05/10
  @Description:Write a function which accepts structure as argument and returns structure to the calling program.
*/
#include <stdio.h>
struct employee
{
 char Name[40];
 char location[40];
 int age;
 int salary;
 char position[40];
};
void read_display(struct employee emp)
{
 printf("Enter information of employee:\n");
 printf("Enter Employee Name:\n");
 scanf(" %[^\n]s", emp.Name);
 printf("Enter Employee location:\n");
 scanf(" %[^\n]s", emp.location);
 printf("Enter Employee position:\n");
 scanf(" %[^\n]s", emp.position);
 printf("Enter Employee age:\n");
 scanf("%d", &emp.age);
 printf("Enter Employee salary:\n");
 scanf("%d", &emp.salary);
 printf("The detail of Employee is :\n");
  printf("Name: %s\tLocation: %s\tPosition: %s\nage: %d\tSalary: %d\n", emp.Name, emp.location, emp.position, emp.age, emp.salary);
}
```

```c
int main()
{
 struct employee emp;
 read_display(emp);

 return 0;
}
```

Output:



Title:

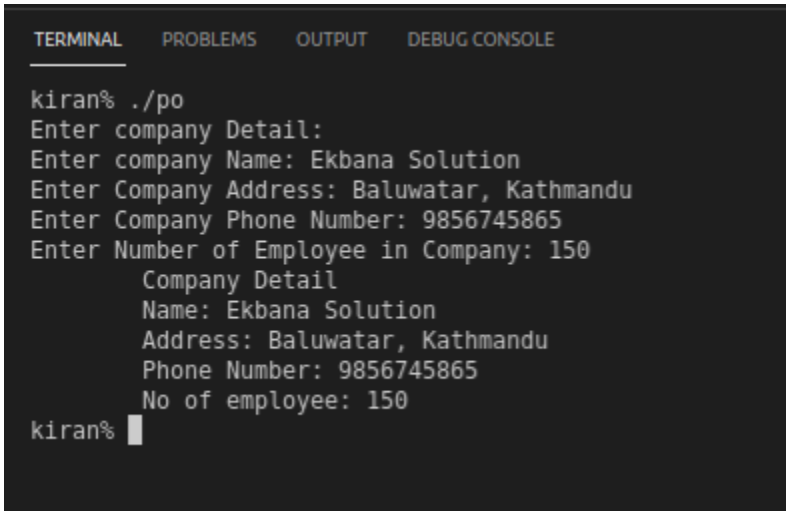
Pass the structures defined in Question 1 into a function and read the structure member and display the values from the function (use structure pointer).

Source Code:

```c
/*/* @Author: Kishan Adhikari
 @Filename: pointerstruct.c
 @Created Date: 2078/05/10
 @Description:Pass the structures defined in Question 1 into a function
and read the structure member and
 display the values from the function (use structure pointer).
*/
#include <stdio.h>
struct company
{
 char name[100];
 char address[100];
 long phone;
 int no_of_Employee;
} comp;

void display(struct company *strtoptr)
{
 printf("\tCompany Detail\t\n");
 printf("\tName: %s\n", strtoptr->name);
 printf("\tAddress: %s\n", strtoptr->address);
 printf("\tPhone Number: %ld\n", strtoptr->phone);
 printf("\tNo of employee: %d\n", strtoptr->no_of_Employee);
}
int main()
{
 struct comp;
 printf("Enter company Detail:\n");
 printf("Enter company Name: ");
 scanf(" %[^\n]s", comp.name);
 printf("Enter Company Address: ");
 scanf(" %[^\n]s", comp.address);
```

```
 printf("Enter Company Phone Number: ");
 scanf("%ld", &comp.phone);
 printf("Enter Number of Employee in Company: ");
 scanf("%d", &comp.no_of_Employee);
 display(&comp);
 return 0;
}
```

Output:



Title:

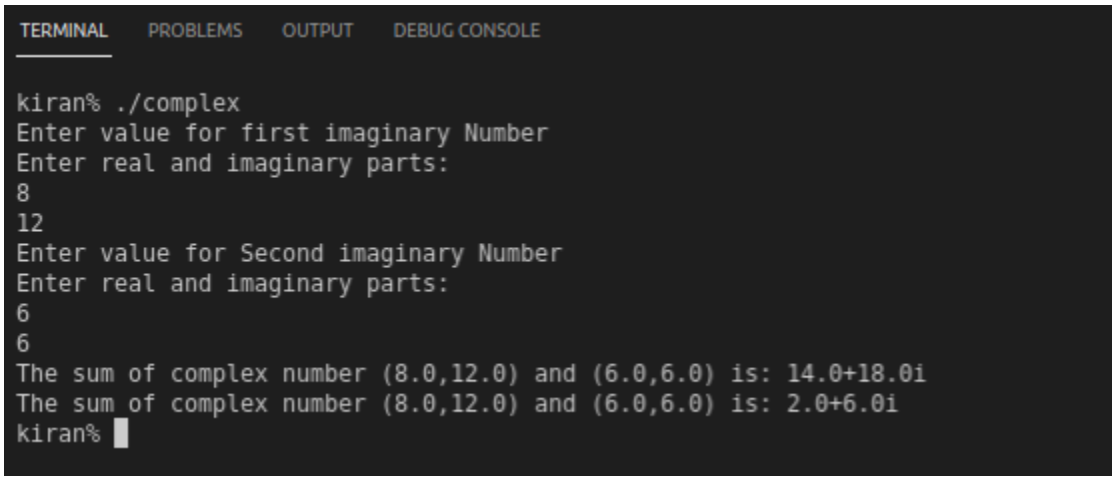
Define a structure “complex” (typedef) to read two complex numbers and perform addition, subtraction of these two complex numbers and display the result.

Source Code:

```
/*@Author: Kishan Adhikari
 @Filename: complex.c
 @Created Date:2078/05/11
 @Description: Define a structure “complex” (typedef) to read two complex numbers and perform addition,subtraction of these two complex numbers and display the result.
*/
```

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

} complex;
void add(complex c1, complex c2)
{
 complex sum;
 sum.real = c1.real + c2.real;
 sum.imaginary = c1.imaginary + c2.imaginary;
  printf("The sum of complex number (%.1f,%.1f) and (%.1f,%.1f) is:
%.1f+%.1fi\n", c1.real, c1.imaginary, c2.real, c2.imaginary, sum.real,
sum.imaginary);
}
void subtract(complex c1, complex c2)
{
 complex sub;
 sub.real = c1.real - c2.real;
 sub.imaginary = c1.imaginary - c2.imaginary;
  printf("The sum of complex number (%.1f,%.1f) and (%.1f,%.1f) is:
%.1f+%.1fi\n", c1.real, c1.imaginary, c2.real, c2.imaginary, sub.real,
sub.imaginary);
}
int main()
{
 complex c1, c2;
 printf("Enter value for first imaginary Number\n");
 printf("Enter real and imaginary parts:\n");
 scanf("%f%f", &c1.real, &c1.imaginary);

 printf("Enter value for Second imaginary Number\n");
 printf("Enter real and imaginary parts:\n");
 scanf("%f%f", &c2.real, &c2.imaginary);
 add(c1, c2);
 subtract(c1, c2);
 return 0;
}
```

Output:



Title:

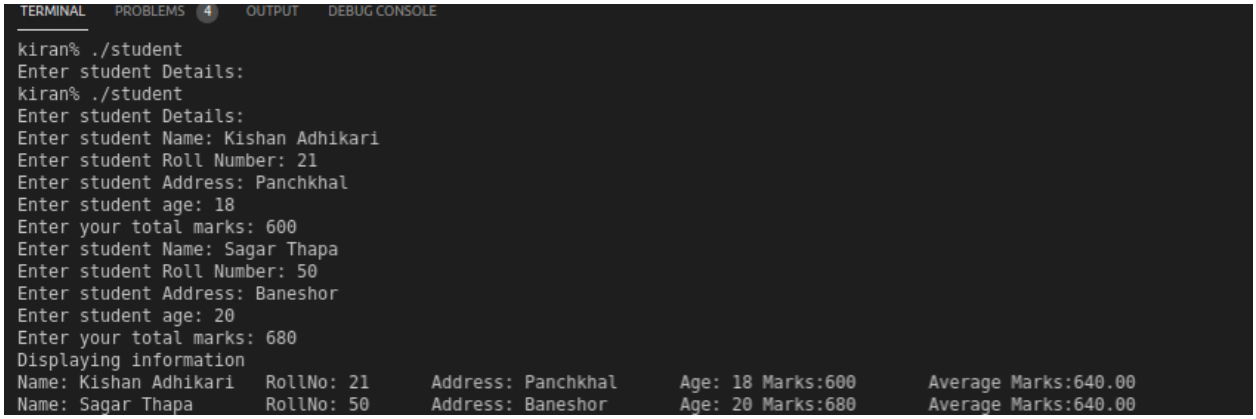
Write a program to read RollNo, Name, Address, Age & average-marks of 12 students in the BCT class and display the details from function.

Source Code:

```
/* @Author: Kishan Adhikari
 @Filename: student.c
 @Created Date: 2078/05/12
  @Description: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
{
 int RollNo, Age;
 char Name[40], Address[40];
 int marks;
} info[2];
```

```c
int total = 0;

float avg;
void display(struct student s[12]);

int main()
{
 struct student info[12];
 int i;
 printf("Enter student Details:\n");
 for (i = 0; i < 12; i++)
 {
   printf("Enter student Name: ");
   scanf(" %[^\n]s", info[i].Name);
   printf("Enter student Roll Number: ");
   scanf("%d", &info[i].RollNo);
   printf("Enter student Address: ");
   scanf(" %[^\n]s", info[i].Address);
   printf("Enter student age: ");
   scanf("%d", &info[i].Age);
   printf("Enter your total marks: ");
   scanf("%d", &info[i].marks);
   total += info[i].marks;
 }
 avg = total / 2;
 display(info);
}

void display(struct student s[12])
{
 printf("Displaying information\n");
 int i;
 for (i = 0; i < 12; i++)
 {
    printf("Name: %s\tRollNo: %d\tAddress: %s\tAge: %d\tMarks:%d\tAverage
Marks:%.2f\n", s[i].Name, s[i].RollNo, s[i].Address, s[i].Age, s[i].marks,
avg);
 }
}
```

Output:



Title:

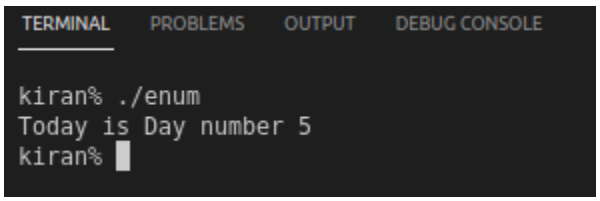
Write a program to show programming examples with union and enumerations.

Source Code #1:

```
/* @Author: Kishan Adhikari
 @Filename: enum.c
 @Created Date: 2078/05/12
  @Description:program to show programming examples with union and enumerations.
*/

#include <stdio.h>
/*enum is a user defined data typein c language. Its syntax is enum{values1,values2,...}
 First element starts with 0 ,second element with 1 and so on
 We can change the default values by declaring values1=4 (value1 has 4 default value)
 */
enum Week
{
```

```
 Sunday = 1,
 Monday,
 Tuesday,
 Wednesday,
 Thursday,
 Friday,
 Saturday
};
int main()
{
 //enum week type variable
 enum Week today;
 today = Thursday;
 printf("Today is Day number %d\n", today);
 return 0;
}
```

Output:

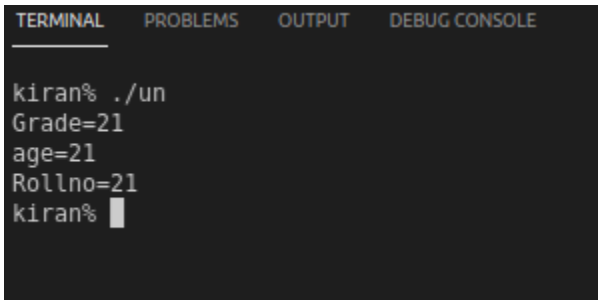


Source Code#2:

```
/* @Author: Kishan Adhikari
 @Filename: enum.c
 @Created Date: 2078/05/12
  @Description:program to show programming examples with union and enumerations.


 Union is another user defined data type.It has similar syntax as that of structure
 It can only handle one data type as it has same storage location for all union members.
 If we define union of Student having member name and age.Then it can only hold one of name and age.
 With a union, all members share the same memory.
```

```
 It is mostly used in embedded system as it lack memory resources
*/

#include <stdio.h>
union student
{
 int grade; //union size is of largest member
 int age;
 int rollno;
};

int main()
{
 union student s;
 s.grade = 9;
 s.age = 12;
 s.rollno = 21;
 printf("Grade=%d\n", s.grade);
 printf("age=%d\n", s.age);
 printf("Rollno=%d\n", s.rollno); //All have same memory and same value
that is last value
 return 0;
}
```

Output:



Title:

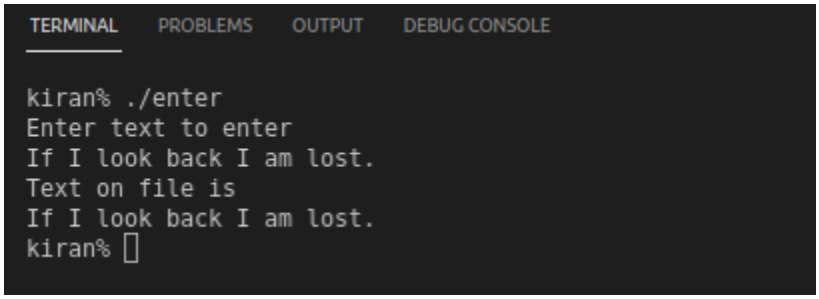
Write characters into a file “filec.txt”. The set of characters are read from the keyboard until an enter key is pressed (use putc() and getc() function).

Source Code:

```c
/* @Author: Kishan Adhikari
 @Filename: enter.c
 @Created Date: 2078/05/10
  @Description:Write characters into a file “filec.txt”. The set of characters are read from the keyboard until
 an enter key is pressed (use putc() and getc() function).
*/
#include <stdio.h> //linking section
#include <stdlib.h>

//main function

int main()
{
 char ch;
 FILE *fptr;
 fptr = fopen("filec.txt", "w+");
 if (fptr == NULL)
 {
   printf("Error opening file");
   exit(0);
 }
 printf("Enter text to enter\n");
 while ((ch = getchar()) != '\n')
 {

   putc(ch, fptr);
 }
 fclose(fptr);
 printf("Text on file is\n");
 fptr = fopen("filec.txt", "r");
 while ((ch = getc(fptr)) != EOF)
 {                    // reading each character from file
   printf("%c", ch); // displaying each character on to the screen
 }
 printf("\n");
 fclose(fptr);
 return 0;
```

```
}
```

Output:



Title:

Read characters from file “filec.txt” created in question 1. Also count the number of characters in the file (use fputs() and fgets() function).

Source Code:

```
/* @Author: Kishan Adhikari
 @Filename: company.c
 @Created Date: 2078/05/10
 @Description:Read characters from file “filec.txt” created in question 1.
 Also count the number of characters in the file (use fputs() and fgets()
function).
*/

#include <stdio.h>
int main()
{
 FILE *fptr;
 int count = 0;
 char ch;
 fptr = fopen("filec.txt", "r");
 if (fptr == NULL)
 {
```

```
   printf("Error opening File");
   return -1;
 }
 while ((ch = getc(fptr)) != EOF) //we consider white space as a character
 {
   count++;
 }
 fclose(fptr);
 printf("Total number of character is %d:\n", count);
}
```

Output:

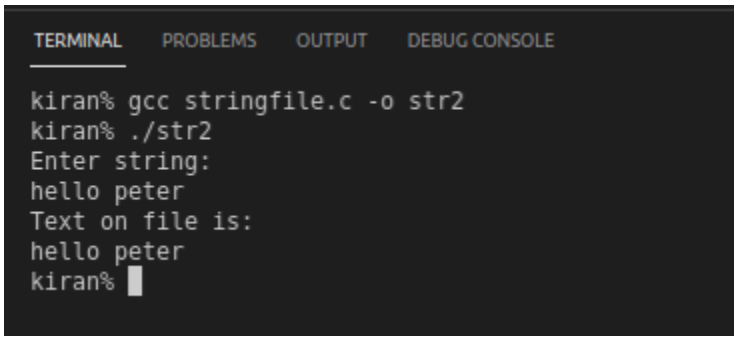
Title:
Write set of strings each of length 40 into a file “stringc.txt” and display it (use fputs() and fgets() function).

Source Code:

```
/* @Author: Kishan Adhikari
 @Filename: structandfunc.c
 @Created Date: 2078/05/10
  @Description:Write set of strings each of length 40 into a file
“stringc.txt” and display it (use fputs() and fgets() function).
*/

#include <stdio.h>
int main()
{
 char str[40];

 FILE *file_ptr;
 file_ptr = fopen("stringc.txt", "w+");

 printf("Enter string:\n");
 fgets(str, 40, stdin);
 fputs(str, file_ptr);
 fclose(file_ptr);
 FILE *fptr = fopen("stringc.txt", "r");
 char line[500];
```

```
 printf("Text on file is:\n");
 while (fgets(line, sizeof(line), fptr))
 {
   printf("%s", line);
 }
 fclose(fptr);
 return 0;
}

*/

#include <stdio.h>
int main()
{
 char str[40];

 FILE *file_ptr;
 file_ptr = fopen("stringc.txt", "w+");

 printf("Enter string:\n");
 fgets(str, 40, stdin);
 fputs(str, file_ptr);
 fclose(file_ptr);
 FILE *fptr = fopen("stringc.txt", "r");
 char line[500];
 printf("Text on file is:\n");
 while (fgets(line, sizeof(line), fptr))
 {
   printf("%s", line);
 }
 fclose(fptr);
 return 0;
}
```

Output:



Title:

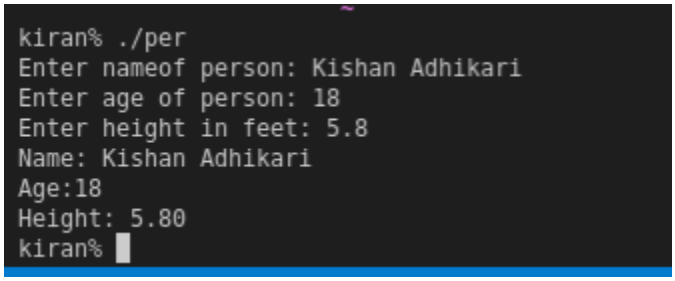
Write name, age and height of a person into a data file “person.txt” and read it (use fprintf() and fscanf() function)

Source Code:

```
/* @Author: Kishan Adhikari
 @Filename: person.c
 @Created Date: 2078/05/20
  @Description:Write name, age and height of a person into a data file
“person.txt” and read it (use fprintf()
 and fscanf() function).
*/
#include <stdio.h>
#include <stdlib.h>
struct person
{
 char name[50];
 int age;
 float height;
} per;
int main()
{
 FILE *file;
 file = fopen("person.txt", "w");
 if (file == NULL)
 {
   printf("Error opening File");
   exit(0);
```

```
 }
 printf("Enter nameof person: ");
 scanf(" %[^\n]s", per.name);
 printf("Enter age of person: ");
 scanf("%d", &per.age);
 printf("Enter height in feet: ");
 scanf("%f", &per.height);
 fprintf("%s %d% %f", per.name, per.age, per.height);
 fclose(file);
 FILE *ptr;
 fscanf(file, "%s %d %f", &per.name, &per.age, per.height);

    printf("Name:    %s\nAge:%d\nHeight:    %.2f\n",    per.name,    per.age,
per.height);

 return 0;
}
```

Output:



Title:

Write a program to print following pattern:

U N
U N I V
U N I V E R
U N I V E R S I
U N I V E R S I T Y
U N I V E R S I
U N I V E R
U N I V
U N

Source Code:

```
/* @Author: Kishan Adhikari
  @Filename: patt.c
  @Description: Program to print following pattern:
 U N
 U N I V
 U N I V E R
 U N I V E R S I
 U N I V E R S I T Y
 U N I V E R S I
 U N I V E R
 U N I V
 U N


*/
#include <stdio.h>
int main()
{
 char word[] = "UNIVERSITY";
 int i, j, n = 2, k = 8;
 for (i = 0; i < 5; i++) //print 5 row
 {
   for (j = 0; j < n; j++)
   {
     printf("%c ", word[j]);
```

```
    }
    n += 2;
    printf("\n");
  }
  for (i = 6; i < 10; i++)
  {
    for (j = 0; j < k; j++)
    {
      printf("%c ", word[j]);
    }
    k -= 2;
    printf("\n");
  }
  return 0;
}
```

Output: