A Laboratory Manual for

Programming for Problem Solving (3110003)

B.E. Semester 2
(Computer Engineering)

Directorate of Technical Education, Gandhinagar,
Gujarat

Government Engineering College, Rajkot

Certificate

This is to certify that Mr./Ms. ____ ________ Enrollment No. ________________ of B.E. Semester _____ Computer Engineering of this Institute (GTU Code: _____) has satisfactorily completed the Practical / Tutorial work for the subject **Programming for Problem Solving (3110003)** for the academic year 2022-23.

Place: __________

Date: __________

Name and Sign of Faculty member

Head of the Department

Practical – Course Outcome matrix

Course Outcomes (COs):

Sr. No.	CO statement
CO-1	Formulate algorithm/flowchart for given arithmetic and logical problem
CO-2	Translate algorithm/flowchart into C program using correct syntax and execute it
CO-3	Write programs using conditional, branching, iteration, and recursion
CO-4	Decompose a problem into function
CO-5	Develop an application using the concepts of array, pointer, structure, and file management to solve engineering and/or scientific problems.

Sr. No.	Objective(s) of Experiment	CO1	CO2	CO3	CO4	CO5
1.	Write C Programs for basic understanding of C programming syntax.		√			
2.	Write C Programs to perform arithmetic operations as per the given equations.		√			
3.	Write C Programs to implement various if statements.			√		
4.	Write C Programs to implement various conditional and branching statements.			√		
5.	Implement given programs using Iteration - Part 1.			√		
6.	Implement given programs using Iteration - Part 2.			√		
7.	Write C Programs to print given patterns using iteration.			√		
8.	Write given programs for concepts of array.					√
9.	Write given programs for string manipulation.					√
10.	Write given programs using User Defined Functions.				√	
11.	Write given programs using concepts of Structure.					√
12.	Write given programs using concepts of Pointers.					√
13.	Write given programs using concepts of File handling.					√
14.	Write given programs to generate mathematical series using control statements and functions.			√	√	

Industry Relevant Skills

The following industry relevant competency are expected to be developed in the student by undertaking the practical work of this laboratory.

1. Problem solving skills
2. Representing a problem using algorithm/flowchart
3. Writing a program for solution of the given problem
4. Documentation for the program using comments in C language

Instructions for Students

1. Students are expected to carefully listen to all the theory classes delivered by the faculty members and understand the COs, content of the course, teaching and examination scheme, skill set to be developed etc.
2. Students shall organize the work in the group and make record of all observations.
3. Students shall develop maintenance skill as expected by industries.
4. Student shall attempt to develop related hand-on skills and build confidence.
5. Student shall develop the habits of evolving more ideas, innovations, skills etc. apart from those included in scope of manual.
6. Student shall refer technical magazines and data books.
7. Student should develop a habit of submitting the experimentation work as per the schedule and s/he should be well prepared for the same.

Common Safety Instructions

1. Tempering of hardware is strictly prohibited.
2. Students should shutdown computer and switch off the power supply after completion of the laboratory session.
3. Students should not remove power plug, mouse or key board.
4. Students should not install any software without permission.
5. Students should not store programs on DESKTOP.

Index
(Progressive Assessment Sheet)

Sr. No.	Objective(s) of Experiment	Page No.	Date of performance	Date of submission	Assessment Marks	Sign. of Teacher with date	Remarks
1.	Write C Programs for basic understanding of C programming syntax.						
2.	Write C Programs to perform arithmetic operations as per the given equations.						
3.	Write C Programs to implement various if statements.						
4.	Write C Programs to implement various conditional and branching statements.						
5.	Implement given programs using Iteration - Part 1.						
6.	Implement given programs using Iteration - Part 2.						
7.	Write C Programs to print given patterns using iteration.						
8.	Write given programs for concepts of array.						
9.	Write given programs for string manipulation.						
10.	Write given programs using User Defined Functions.						
11.	Write given programs using concepts of Structure.						
12.	Write given programs using concepts of Pointers.						
13.	Write given programs using concepts of File handling.						
14.	Write given programs to generate mathematical series using control statements and functions.						
Total							

Experiment – 1

Write C Programs for basic understanding of C programming syntax.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

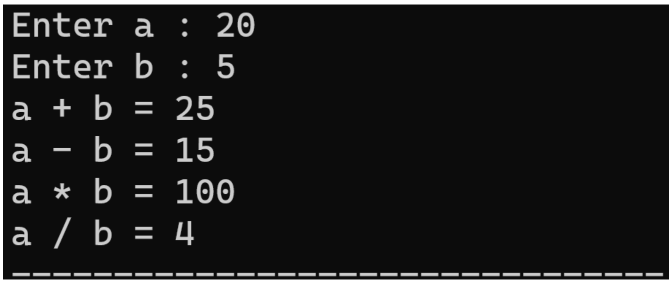
Objectives: (a) To understand the structure and syntax of C Program.

(b) To understand declaration and use of variables in C Program.

Equipment/Instruments: Computer with C Compiler

1) Write a program to that performs as calculator (addition, multiplication, division, subtraction).

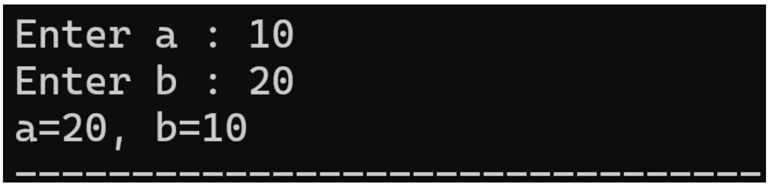
```
#include<stdio.h>

int main()
{
        int a,b,ans;

        printf("Enter a : ");
        scanf("%d",&a);

        printf("Enter b : ");
        scanf("%d",&b);

        ans = a + b;
        printf("a + b = %d \n",ans);

        ans = a - b;
        printf("a - b = %d \n",ans);

        ans = a * b;
        printf("a * b = %d \n",ans);

        ans = a / b;
        printf("a / b = %d ",ans);

        return 0;
}
```



```
Enter a : 20
Enter b : 5
a + b = 25
a - b = 15
a * b = 100
a / b = 4
--------------------------------
```

2) Write a C program to interchange two numbers.

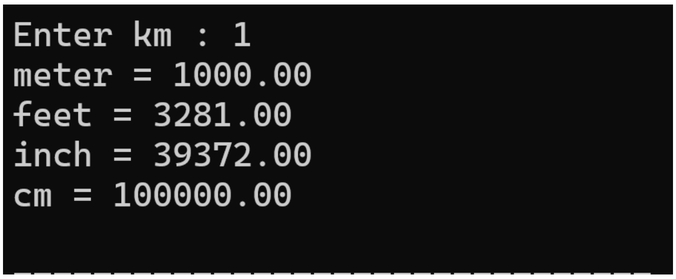
```
#include<stdio.h>

void main()
{
        int a,b,Temp;

        printf("Enter a : ");
        scanf("%d",&a);
        printf("Enter b : ");
        scanf("%d",&b);

        Temp=a;
        a=b;
        b=Temp;

        printf("a=%d, b=%d ",a,b);
}
```



```
Enter a : 10
Enter b : 20
a=20, b=10
--------------------------------
```

3) Write a C program to enter a distance into kilometer and convert it in to meter, feet, inches and centimeter.

```
#include<stdio.h>
void main()
{
        float km,meter,feet,inch,cm;

        printf("Enter km : ");
        scanf("%f",&km);

        meter = km * 1000;

        feet = meter * 3.281;

        inch = feet * 12;

        cm = km * 100000;

        printf("meter = %0.2f \n",meter);
        printf("feet = %0.2f \n",feet);
        printf("inch = %0.2f \n",inch);
        printf("cm = %0.2f \n",cm);
}
```



```
Enter km : 1
meter = 1000.00
feet = 3281.00
inch = 39372.00
cm = 100000.00
```

Experiment – 2

Write C Programs to perform arithmetic operations as per the given equations.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

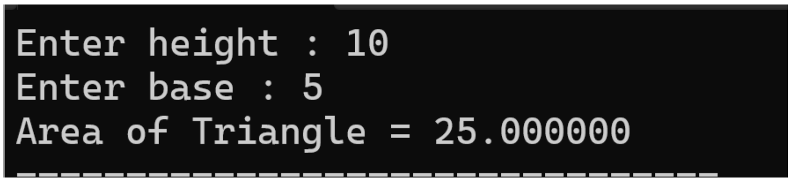
Objectives: (a) To understand the implementation of arithmetic expressions.

(b) To understand declaration and use of variables in C Program.

Equipment/Instruments: Computer with C Compiler

1) Write a program to find area of triangle (a=h*b*.5) a = area, h = height, b = base

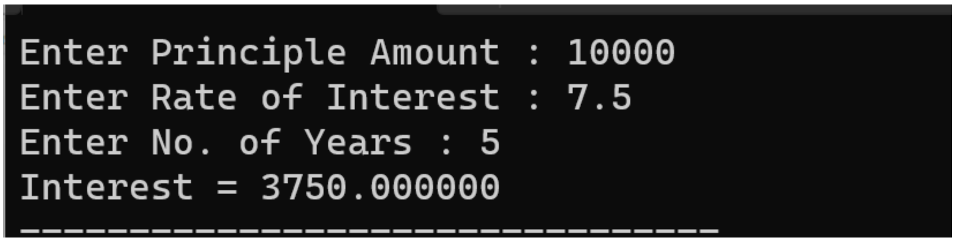
```
#include<stdio.h>
void main()
{
        float a,h,b;

        printf("Enter height : ");
        scanf("%f",&h);
        printf("Enter base : ");
        scanf("%f",&b);

        a=0.5*h*b;

        printf("Area of Triangle = %f ",a);
}
```



```
Enter height : 10
Enter base : 5
Area of Triangle = 25.000000
--------------------------------
```

2) Write a program to calculate simple interest
i = (p*r*n)/100
i = Simple interest
p = Principal amount
r = Rate of interest
n = Number of years

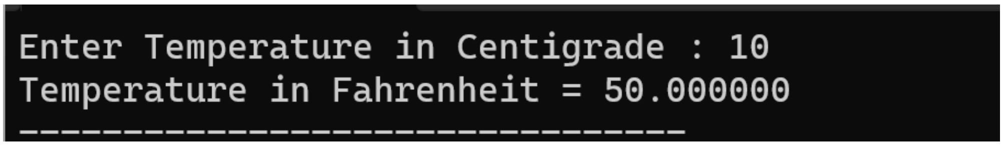
```
/*
i = Simple interest
p = Principal amount
r = Rate of interest
n = Number of years
*/

#include<stdio.h>

void main()
{
        float I,P,R,N;

        printf("Enter Principle Amount : ");
        scanf("%f",&P);
        printf("Enter Rate of Interest : ");
        scanf("%f",&R);
        printf("Enter No. of Years : ");
        scanf("%f",&N);

        I=(P*R*N)/100;

        printf("Interest = %f ",I);
}
```



```
Enter Principle Amount : 10000
Enter Rate of Interest : 7.5
Enter No. of Years : 5
Interest = 3750.000000
--------------------------------
```

3) Write a program to compute Fahrenheit from Celsius. (f=1.8*c +32)

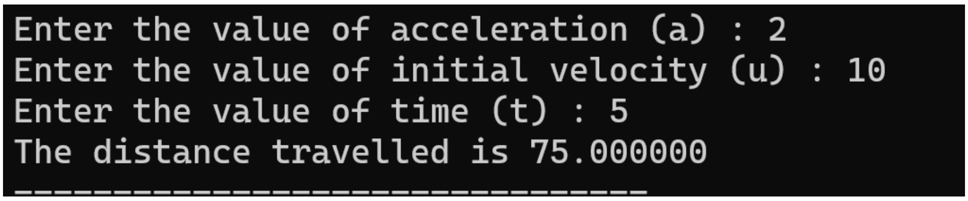
```
void main()
{
        float F,C;

        printf("Enter Temperature in Centigrade : ");
        scanf("%f",&C);

        F = 1.8 * C + 32;

        printf("Temperature in Fahrenheit = %f ", F);
}
```



```
Enter Temperature in Centigrade : 10
Temperature in Fahrenheit = 50.000000
--------------------------------
```

4) Write a C program to find out distance travelled by the equation d = ut + ½ at^2

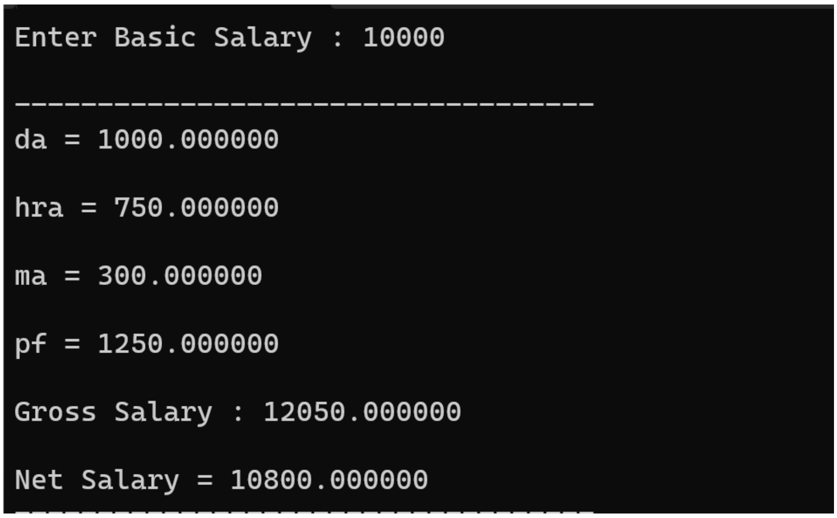
```
/*
d = distance
u = initial velocity
a = acceleration
t = time
*/

#include<stdio.h>
void main()
{
   float u,a,d;
   int t;

   printf("Enter the value of acceleration (a) : ");
   scanf("%f",&a);

   printf("Enter the value of initial velocity (u) : ");
   scanf("%f",&u);

   printf("Enter the value of time (t) : ");
   scanf("%d",&t);

   // d = ut + (at^2)/2

   d = (u * t) + (a * t * t)/2;

   printf("The distance travelled is %f ",d);

}
```



```
Enter the value of acceleration (a) : 2
Enter the value of initial velocity (u) : 10
Enter the value of time (t) : 5
The distance travelled is 75.000000
--------------------------------
```

5) Write a c program to prepare pay slip using following data:
Da = 10% of basic,
Hra = 7.50% of basic,
Ma = 300,
Pf = 12.50% of basic,
Gross = basic + Da + Hra + Ma,
Nt = Gross – Pf.

```
#include<stdio.h>
void main()
{
        float basic,da,hra,ma,pf,gross,net;

        printf("Enter Basic Salary : ");
        scanf("%f",&basic);

        da = (basic * 10) / 100;

        hra= (basic * 7.5) / 100;

        ma = 300;

        pf = (basic * 12.5) / 100;

        gross = basic + da + hra + ma;

        net = gross - pf;

        printf("\n-----------------------------------\n");
        printf("da = %f \n\n",da);
        printf("hra = %f \n\n", hra);
        printf("ma = %f \n\n",ma);
        printf("pf = %f \n\n",pf);
        printf("Gross Salary : %f \n\n",gross);
        printf("Net Salary = %f",net);
        printf("\n-----------------------------------\n");
}
```



```
Enter Basic Salary : 10000

-----------------------------------
da = 1000.000000

hra = 750.000000

ma = 300.000000

pf = 1250.000000

Gross Salary : 12050.000000

Net Salary = 10800.000000
-----------------------------------
```

Experiment – 3

Write C Programs to implement various if statements.

Date:

Competency and Practical Skills: C basic programming syntax and if statements

Relevant CO: CO-2

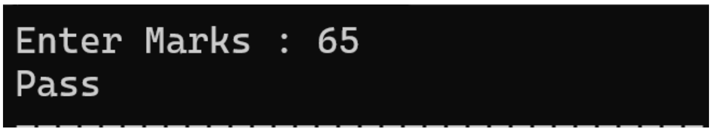
Objective: (a) To understand the syntax and implementation of various if statements.

(b) To understand the use of various if statements according to the requirement.

Equipment/Instruments: Computer with C Compiler

1) Write a program to read marks of a student from keyboard and check whether the student is pass or fail (using if else).

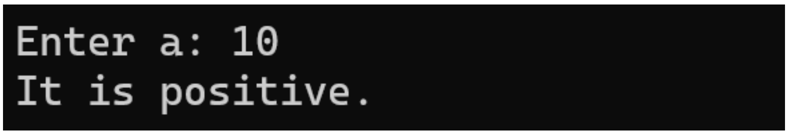
```
#include<stdio.h>
void main()
{
	int Marks;

	printf("Enter Marks : ");
	scanf("%d",&Marks);

	if(Marks >= 40)
	{
		printf("Pass");
	}
	else
	{
		printf("Fail");
	}
}
```



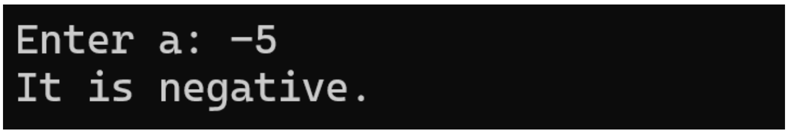
```
Enter Marks : 65
Pass
```

2) Write a C program to find that the accepted number is Negative, or Positive or Zero.

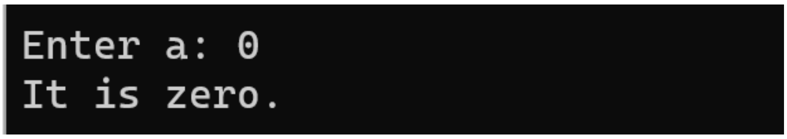
```
#include <stdio.h>
void main()
{
        int a;

        printf("Enter a: ");
        scanf("%d", &a);

        if (a > 0)
        {
            printf("It is positive.");
        }
        else if (a < 0)
        {
            printf("It is negative.");
        }
        else
        {
            printf("It is zero.");
        }
}
```



```
Enter a: 10
It is positive.
```



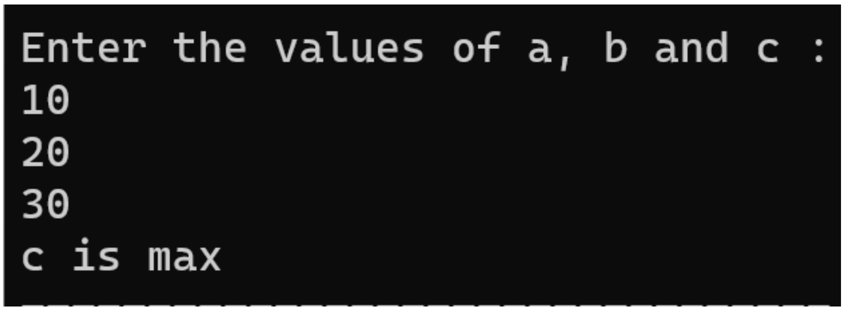
```
Enter a: -5
It is negative.
```



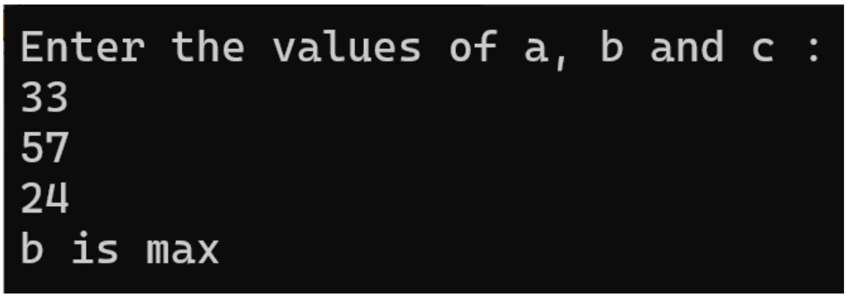
```
Enter a: 0
It is zero.
```

3) Write a program to read three numbers from keyboard and find out maximum out of these three. (nested if else)

```
#include <stdio.h>
void main()
{
    int a, b, c;
    printf("Enter the values of a, b and c : \n");
    scanf("%d%d%d",&a,&b,&c);

    if (a > b)
    {
            if (a > c)
            {
                        printf("a is max");
            }
            else
            {
                        printf("c is max");
            }
    }
    else
    {
            if (b > c)
            {
                        printf("b is max");
            }
            else
            {
                       printf("c is max");
            }
     }
}
```



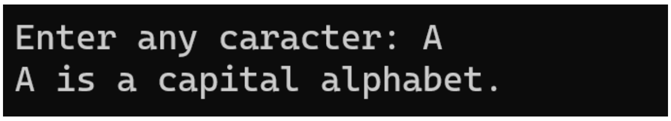
```
Enter the values of a, b and c :
10
20
30
c is max
```



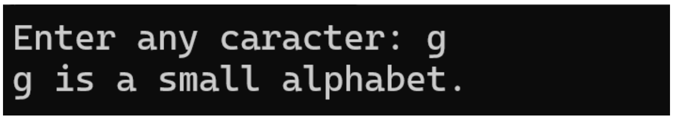
```
Enter the values of a, b and c :
33
57
24
b is max
```

4) Write a C program to check whether the entered character is capital, small letter, digit or any special character.

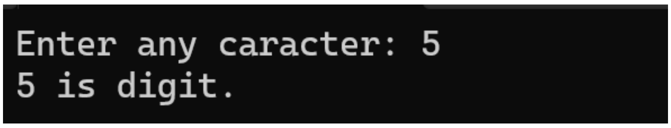
```
#include <stdio.h>
void main()
{
		char c;

		printf("Enter any caracter: ");
		scanf("%c", &c);

		if(c >= 'a' && c <= 'z')
		{
			printf("%c is a small alphabet.", c);
		}
		else if(c >= 'A' && c <= 'Z')
		{
			printf("%c is a capital alphabet.", c);
		}
		else if(c >= '0' && c <= '9')
		{
			printf("%c is digit.", c);
		}
		else
		{
			printf("%c is special caracter.", c);
		}
}
```



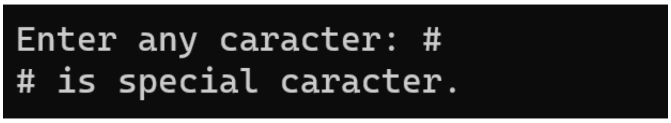
```
Enter any caracter: A
A is a capital alphabet.
```



```
Enter any caracter: g
g is a small alphabet.
```



```
Enter any caracter: 5
5 is digit.
```



```
Enter any caracter: #
# is special caracter.
```

Experiment – 4

Write C Programs to implement various conditional and branching statements**.**

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

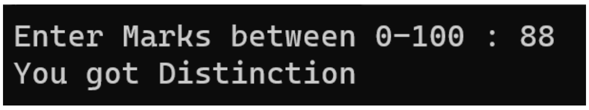
Objectives: (a) To understand the syntax of various conditional and branching statements.
(b) To understand the use of various conditional and branching statements according to the requirement.

Equipment/Instruments: Computer with C Compiler

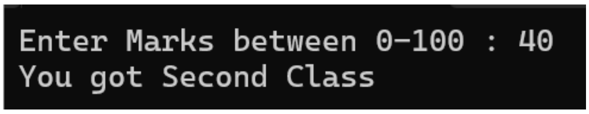
1) Write a program to read marks from keyboard and your program should display equivalent grade according to following table (if else ladder)

Marks	**Grade**
100 - 80	**Distinction**
79 - 60	**First Class**
59 - 40	**Second Class**
< 40	**Fail**

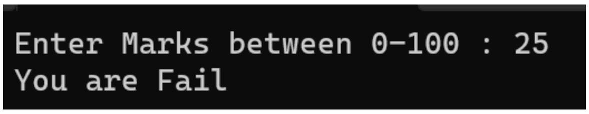
```
#include<stdio.h>
void main()
{
        int n;
        printf("Enter Marks between 0-100 : ");
        scanf("%d",&n);

        if(n>100 || n < 0)
        {
                printf("Your Input is out of Range");
                exit(0);
        }

        if(n>=80)
        {
                printf("You got Distinction");
        }
        else if(n>=60)
        {
                printf("You got First Class");
        }
        else if(n>=40)
        {
                printf("You got Second Class");
        }
        else
        {
                printf("You are Fail");
        }

}
```



```
Enter Marks between 0-100 : 88
You got Distinction
```



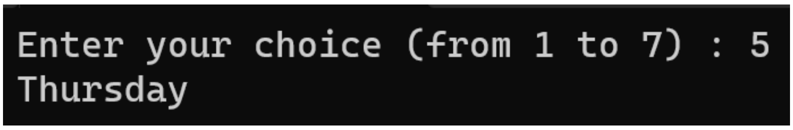
```
Enter Marks between 0-100 : 40
You got Second Class
```



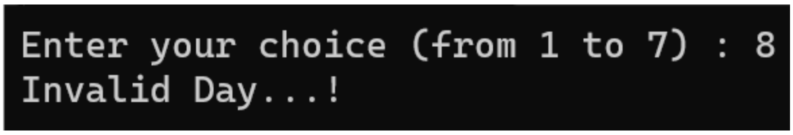
```
Enter Marks between 0-100 : 25
You are Fail
```

2) Write a C program to read no 1 to 7 and print relatively day Sunday to Saturday.

```
#include<stdio.h>
void main()
{
        int choice;

        printf("Enter your choice (from 1 to 7) : ");
        scanf("%d",&choice);

        switch(choice)
        {
                case 1:
                        printf("Sunday");
                        break;
                case 2:
                        printf("Monday");
                        break;
                case 3:
                        printf("Tuesday");
                        break;
                case 4:
                        printf("Wednesday");
                        break;
                case 5:
                        printf("Thursday");
                        break;
                case 6:
                        printf("Friday");
                        break;
                case 7:
                        printf("Saturday");
                        break;
                default:
                        printf("Invalid Day...!");
        }
}
```



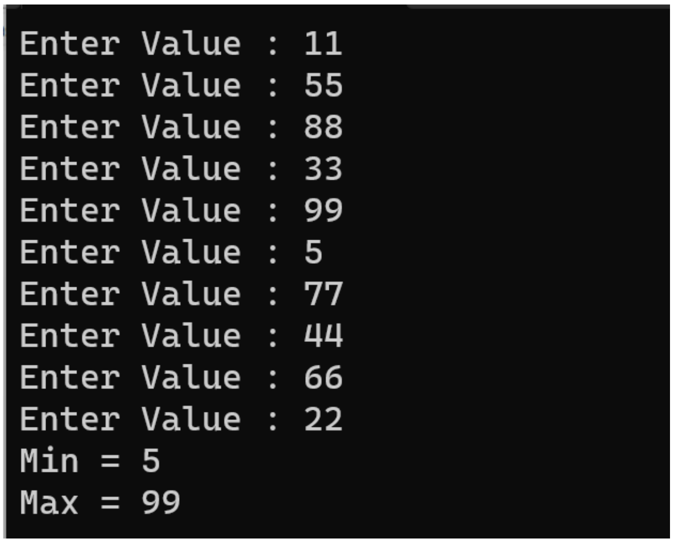
```
Enter your choice (from 1 to 7) : 5
Thursday
```



```
Enter your choice (from 1 to 7) : 8
Invalid Day...!
```

3) Write a C program to find out the Maximum and Minimum number from given 10 numbers.

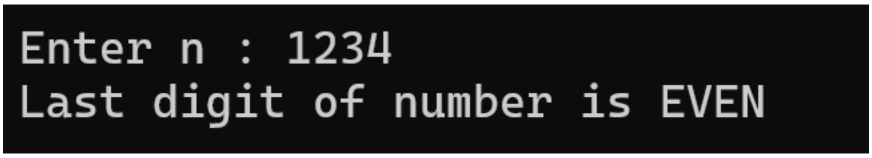
```
#include<stdio.h>
void main()
{
        int a,i,max,min;

        for(i=1 ; i<=10 ; i++)
        {
                printf("Enter Value : ");
                scanf("%d",&a);

                if(i==1)
                {
                        min = a;
                        max = a;
                }
                else
                {
                        if(a > max)
                        {
                                max = a;
                        }
                        if(a < min)
                        {
                                min = a;
                        }
                }
        }

        printf("Min = %d  \n", min);
        printf("Max = %d", max);
}
```



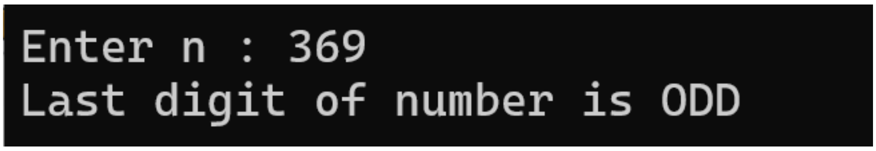
```
Enter Value : 11
Enter Value : 55
Enter Value : 88
Enter Value : 33
Enter Value : 99
Enter Value : 5
Enter Value : 77
Enter Value : 44
Enter Value : 66
Enter Value : 22
Min = 5
Max = 99
```

4) Write a C program to input an integer number and check the last digit of number is even or odd.

```
#include<stdio.h>
void main()
{
	int n;

	printf("Enter n : ");
	scanf("%d",&n);

	if(n%2 == 0)
	{
		printf("Last digit of number is EVEN");
	}
	else
	{
		printf("Last digit of number is ODD");
	}
}
```



```
Enter n : 1234
Last digit of number is EVEN
```



```
Enter n : 369
Last digit of number is ODD
```

Experiment – 5

Implement given programs using Iteration - Part 1.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

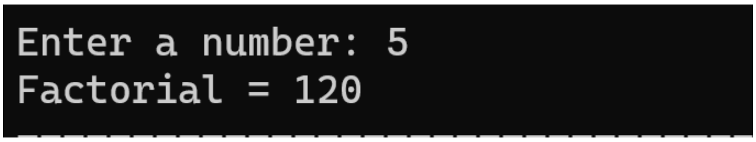
Relevant CO: CO-2

Objective: (a) To understand the use of iterative statements to solve given problems.

Equipment/Instruments: Computer with C Compiler

1) Write a C program to find factorial of a given number.

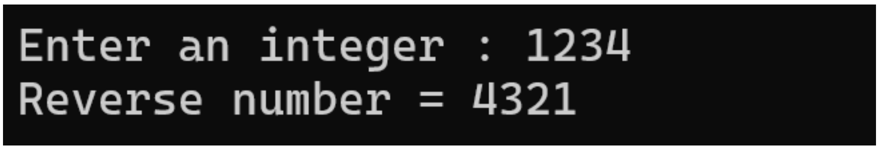
```
#include<stdio.h>
void main()
{

        int i,fact,n;
        printf("Enter a number: ");
        scanf("%d",&n);

        fact=1;
        for(i=1;i<=n;i++)
        {
                fact=fact*i;
        }

        printf("Factorial = %d",fact);
}
```



```
Enter a number: 5
Factorial = 120
```

2) Write a program to reverse a number.

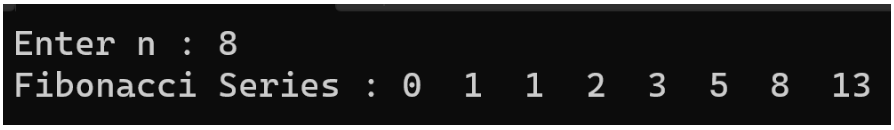
```
#include <stdio.h>
void main()
{
          int n, rev = 0, temp;
          printf("Enter an integer : ");
          scanf("%d", &n);

          while (n != 0)
          {
               temp = n % 10;
               rev = rev * 10 + temp;
               n = n / 10;
          }
          printf("Reverse number = %d", rev);
}
```



```
Enter an integer : 1234
Reverse number = 4321
```

3) Write a program to generate first n number of Fibonacci series

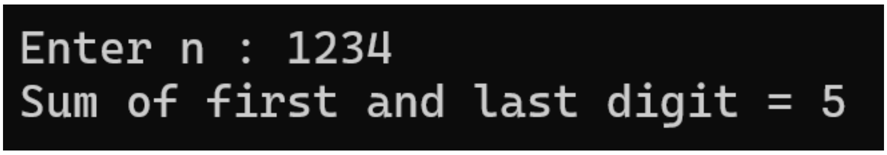
```
#include<stdio.h>

void main()
{
	int n,a,b,c,count;

	printf("Enter n : ");
	scanf("%d",&n);

	a=0;
	b=1;

	printf("Fibonacci Series : %d  %d  ",a,b);

	count = 2;

	while(count < n)
	{
		c = a + b;
		printf("%d  ",c);
		count++;

		a = b;
		b = c;
	}
}
```



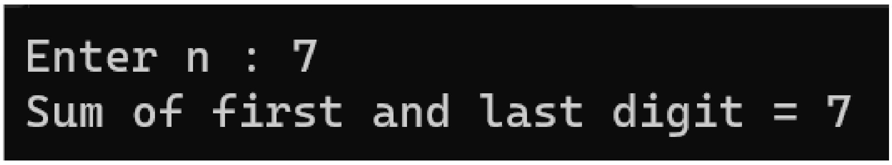
```
Enter n : 8
Fibonacci Series : 0  1  1  2  3  5  8  13
```

4) Write a program to find out sum of first and last digit of a given number.

```
#include<stdio.h>
void main()
{
	int n,first,last;

	printf("Enter n : ");
	scanf("%d",&n);

	if(n<=9)
	{
		printf("Sum of first and last digit = %d ", n);
		exit(0);
	}

	last = n%10;

	while(n>9)
	{
		n = n/10;
	}

	first=n;

	printf("Sum of first and last digit = %d ", first+last);
}
```



```
Enter n : 1234
Sum of first and last digit = 5
```



```
Enter n : 7
Sum of first and last digit = 7
```

Experiment – 6

Implement given programs using Iteration - Part 2**.**

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

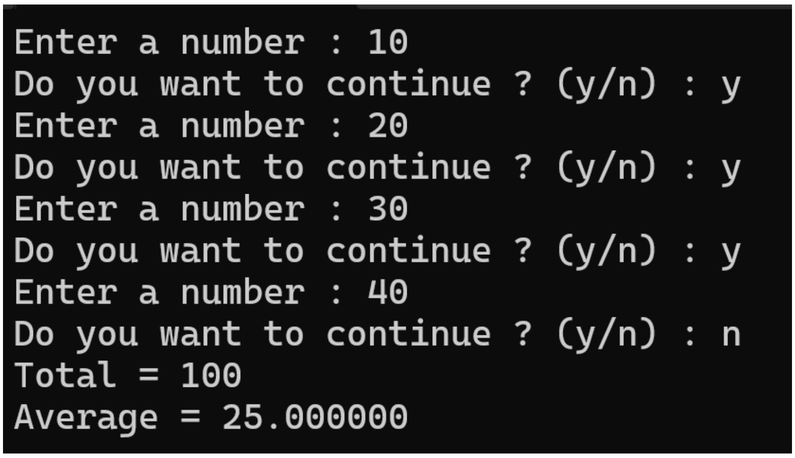
Relevant CO: CO-2

Objective: (a) To understand the use of iterative statements to solve given problems.

Equipment/Instruments: Computer with C Compiler

1) Write a C program to find the sum and average of different numbers which are accepted by user as many as user wants.

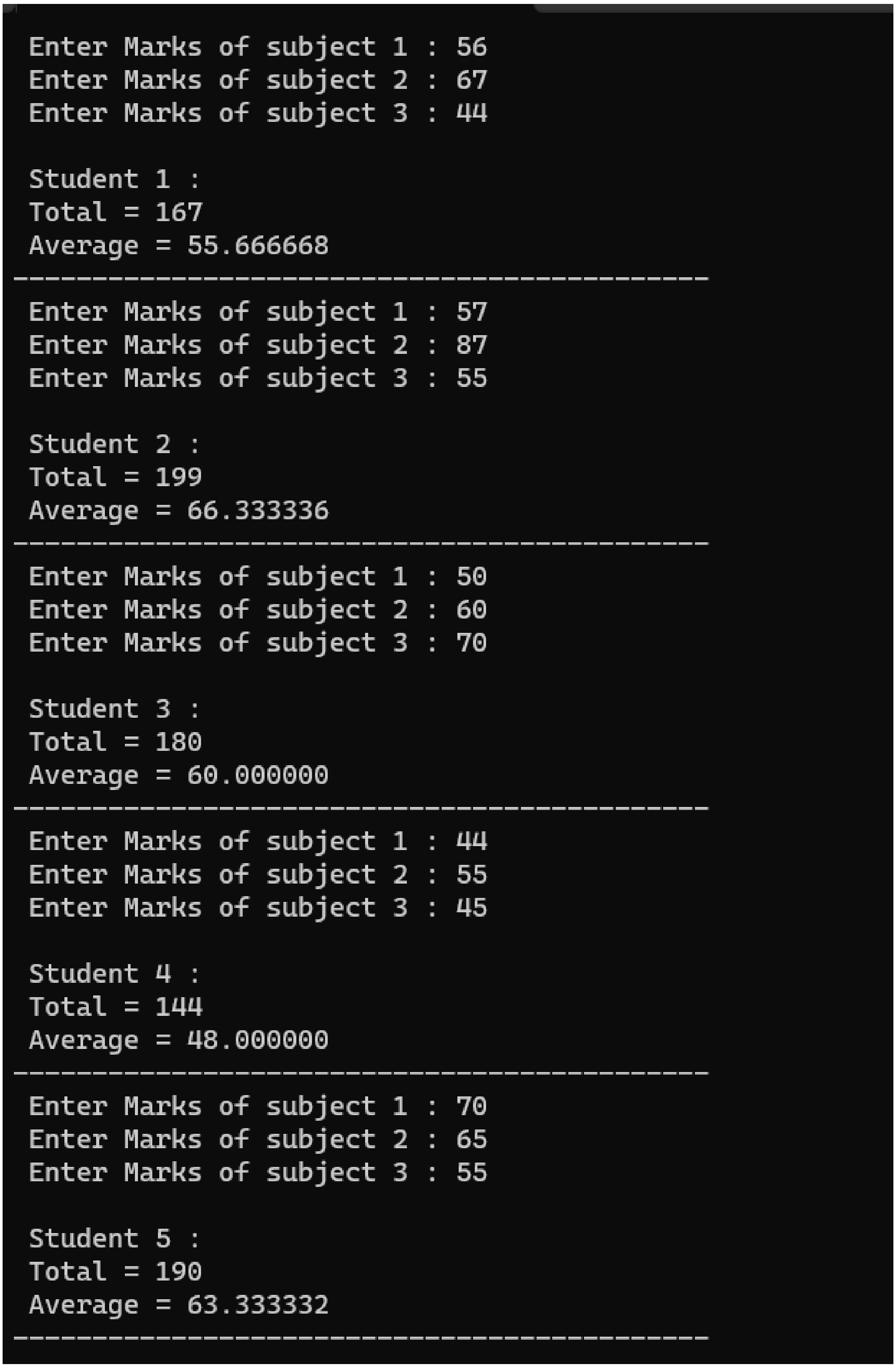
```
#include<stdio.h>

void main()
{
        int n,total=0,count=0;
        float avg;
        char choice;

        do
        {
                printf("Enter a number : ");
                scanf("%d",&n);

                total=total+n;
                count++;

                fflush(stdin);

                printf("Do you want to continue ? (y/n) : ");
                scanf("%c",&choice);

        }while(choice == 'y');

        avg=(float)total/count;

        printf("Total = %d \n",total);
        printf("Average = %f ",avg);
}
```



```
Enter a number : 10
Do you want to continue ? (y/n) : y
Enter a number : 20
Do you want to continue ? (y/n) : y
Enter a number : 30
Do you want to continue ? (y/n) : y
Enter a number : 40
Do you want to continue ? (y/n) : n
Total = 100
Average = 25.000000
```

2) Write a program to calculate average and total of 5 students for 3 subjects (use nested for loops)

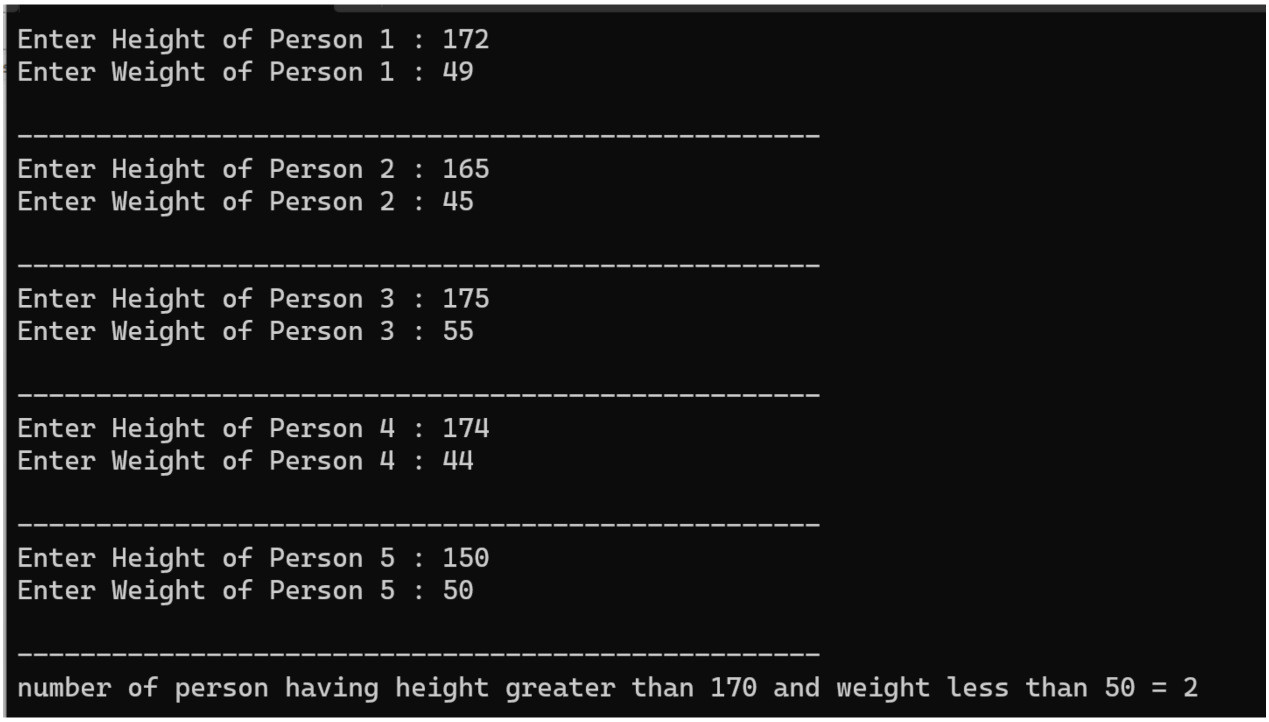
```
#include<stdio.h>
void main()
{
        int m,total=0,i,j;
        float avg;

        for(i=1;i<=5;i++)
        {
                total=0;

                for(j=1;j<=3;j++)
                {
                        printf(" Enter Marks of subject %d : ",j);
                        scanf("%d",&m);
                        total=total + m;
                }

                avg=total/3.0;

                printf("\n Student %d : \n Total = %d \n Average = %f  \n",i,total,avg);

                printf("-------------------------------------------\n");
        }
}
```



```
 Enter Marks of subject 1 : 56
 Enter Marks of subject 2 : 67
 Enter Marks of subject 3 : 44

 Student 1 :
 Total = 167
 Average = 55.666668
--------------------------------------------
 Enter Marks of subject 1 : 57
 Enter Marks of subject 2 : 87
 Enter Marks of subject 3 : 55

 Student 2 :
 Total = 199
 Average = 66.333336
--------------------------------------------
 Enter Marks of subject 1 : 50
 Enter Marks of subject 2 : 60
 Enter Marks of subject 3 : 70

 Student 3 :
 Total = 180
 Average = 60.000000
--------------------------------------------
 Enter Marks of subject 1 : 44
 Enter Marks of subject 2 : 55
 Enter Marks of subject 3 : 45

 Student 4 :
 Total = 144
 Average = 48.000000
--------------------------------------------
 Enter Marks of subject 1 : 70
 Enter Marks of subject 2 : 65
 Enter Marks of subject 3 : 55

 Student 5 :
 Total = 190
 Average = 63.333332
--------------------------------------------
```

3) Read five persons height and weight and count the number of persons having height greater than 170 and weight less than 50.

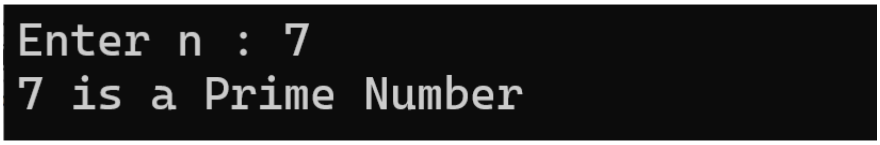
```
#include<stdio.h>
void main()
{
	int i,height,weight,count=0;

	for(i=1;i<=5;i++)
	{
		printf("Enter Height of Person %d : ",i);
		scanf("%d",&height);
		printf("Enter Weight of Person %d : ",i);
		scanf("%d",&weight);

		if(height > 170 && weight < 50)
		{
			count++;
		}

		printf("\n---------------------------------------------------\n");
	}

	printf("number of person having height greater than 170 and weight less than 50 = %d ", count);
}
```



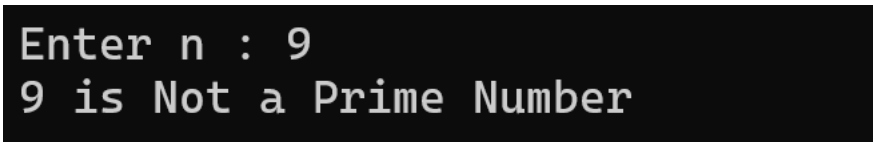
```
Enter Height of Person 1 : 172
Enter Weight of Person 1 : 49

---------------------------------------------------
Enter Height of Person 2 : 165
Enter Weight of Person 2 : 45

---------------------------------------------------
Enter Height of Person 3 : 175
Enter Weight of Person 3 : 55

---------------------------------------------------
Enter Height of Person 4 : 174
Enter Weight of Person 4 : 44

---------------------------------------------------
Enter Height of Person 5 : 150
Enter Weight of Person 5 : 50

---------------------------------------------------
number of person having height greater than 170 and weight less than 50 = 2
```

4) Write a program to check whether the given number is prime or not.

```
#include<stdio.h>
void main()
{
        int n,i,flag=0;

        printf("Enter n : ");
        scanf("%d",&n);

        for(i=2 ; i<=n-1 ; i++)
        {
                if(n%i == 0)
                {
                        flag = 1;
                        break;
                }
        }

        if(flag==1)
        {
                printf("%d is Not a Prime Number ",n);
        }
        if(flag==0)
        {
                printf("%d is a Prime Number  ",n);
        }
}
```



```
Enter n : 7
7 is a Prime Number
```



```
Enter n : 9
9 is Not a Prime Number
```

Experiment – 7

Write C Programs to print given patterns using iteration**.**

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

Objective: (a) To understand the use of iterative statements to print given pattens.

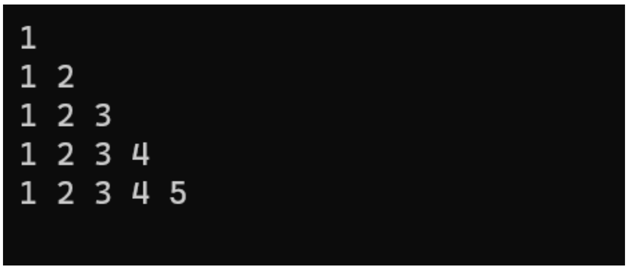
Equipment/Instruments: Computer with C Compiler

1) Write a program to print following patterns:

a)

1
22
333
4444
55555

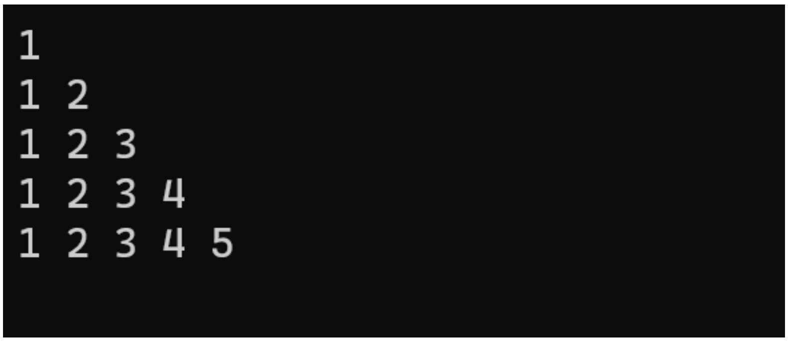
```
#include<stdio.h>
void main()
{
	int I,j;

	for(i=1 ; i<=5 ; i++ )
	{
		for(j=1 ; j<=I ; j++ )
		{
			printf("%d ",j);
		}
		printf("\n");
	}
}
```



```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

b)
1
12
123
1234
12345

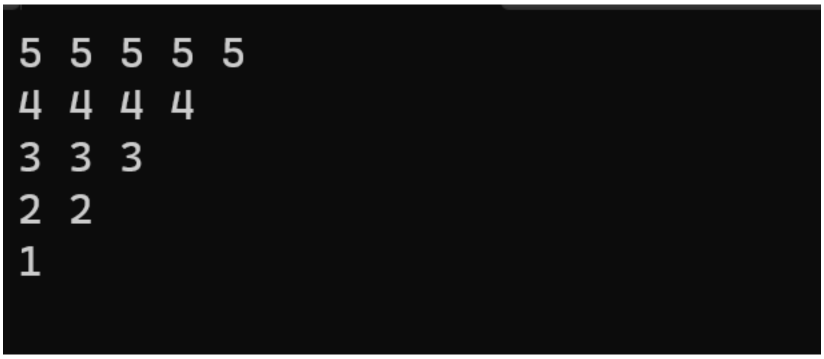
```
#include<stdio.h>
void main()
{
    int I,j;

    for( i=1 ; i<=5 ; i++ )
    {
            for(j=1 ; j<=I ; j++ )
            {
                    printf(“%d “,j);
            }
            printf(“\n”);
    }
}
```



```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

c)
55555
4444
333
22
1

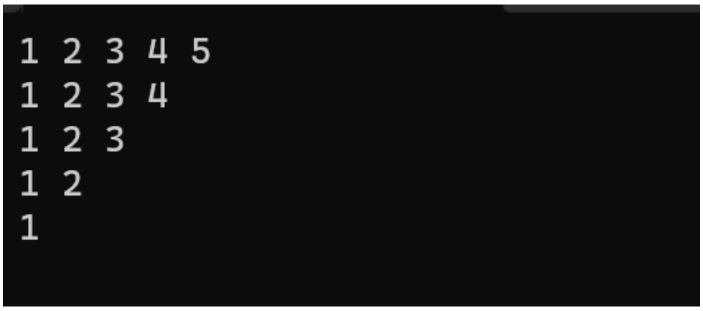
```
#include <stdio.h>
void main()
{
    int I,j;

    for(i=5;i>=1;i--)
    {
        for(j=1;j<=I;j++)
        {
            printf(“%d “,i);
        }
        printf(“\n”);
    }
}
```



```
5 5 5 5 5
4 4 4 4
3 3 3
2 2
1
```

d)
12345
1234
123
12
1

```
#include <stdio.h>
void main()
{
        int I,j;

        for(i=5;i>=1;i--)
         {
                for(j=1;j<=I;j++)
                {
                        printf("%d ",j);
                }
                printf("\n");
        }
}
```



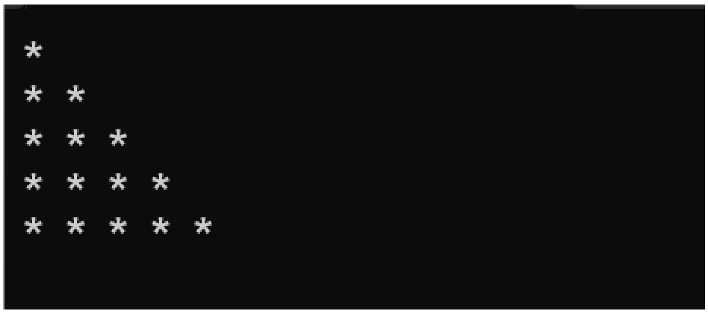
```
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

2) Write a program to print following patterns:

a)

```
*
* *
* * *
* * * *
* * * * *
```

```
#include <stdio.h>
void main()
{
        int I,j;

        for(i=1;i<=5;i++)
        {
                for(j=1;j<=I;j++)
                {
                        printf(“* “);
                }
                printf(“\n”);
        }
}
```

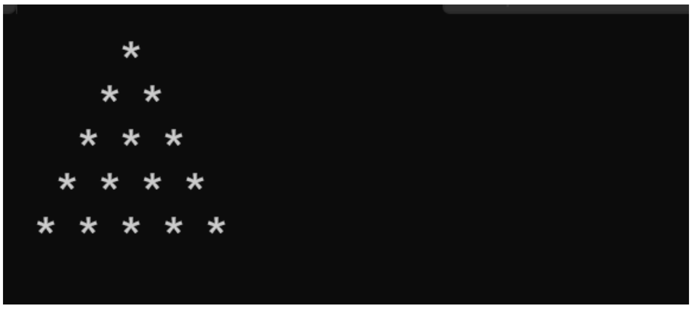


```
*
* *
* * *
* * * *
* * * * *
```

b)

```
      *
     * *
    * * *
   * * * *
  * * * * *
```

```
#include <stdio.h>
void main()
{
        int I,j,k;

        for(i=1;i<=5;i++)
        {
                for(k=1; k<= 6-I; k++ )
                {
                        printf(“ “);
                }

                for(j=1;j<=I;j++)
                {
                        printf(“* “);
                }
                printf(“\n”);
         }
}
```

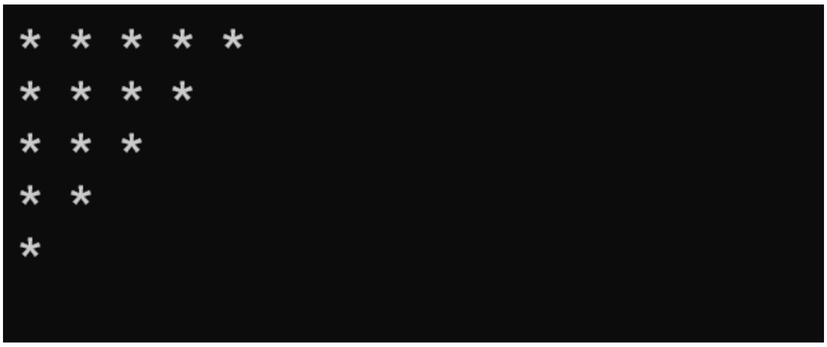


```
    *
   * *
  * * *
 * * * *
* * * * *
```

c)

```
* * * * *
* * * *
* * *
* *
*
```

```
#include <stdio.h>
void main()
{
        int I,j;

        for(i=5;i>=1;i--)
         {
                for(j=1;j<=I;j++)
                {
                        printf("* ");
                }
                printf("\n");
         }
}
```



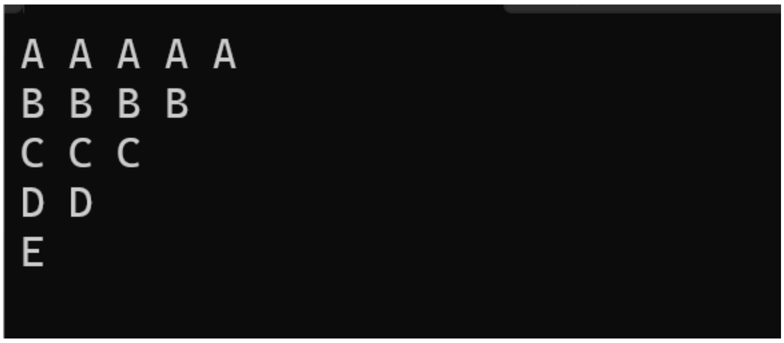
```
* * * * *
* * * *
* * *
* *
*
```

3) Write a program to print following patterns:

a)

AAAAA
BBBB
CCC
DD
E

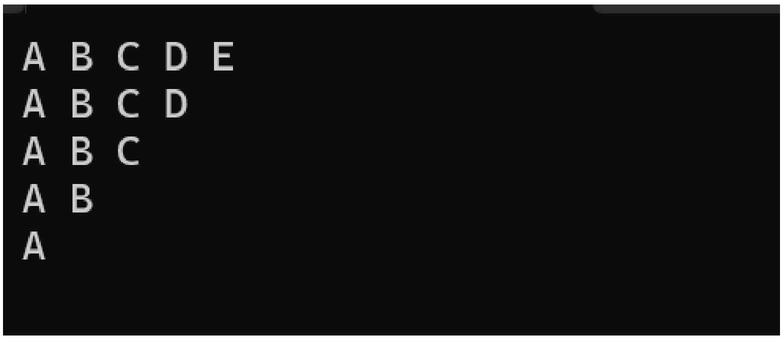
```
#include <stdio.h>
void main()
{
          int I,j;
           char c;

           c='A';

            for(i=5 ; i>=1 ; i--)
           {
                    for(j=I ; j>=1 ; j--)
                    {
                              printf("%c ", c);
                    }
                    printf("\n");
                    c++;
          }
}
```



```
A A A A A
B B B B
C C C
D D
E
```

b)
ABCDE
ABCD
ABC
AB
A

```
#include <stdio.h>
void main()
{
        int I,j;
        char c;

        c='A';

        for(i=5;i>=1;i--)
        {
                c='A';
                for(j=I;j>=1;j--)
                {
                        printf("%c ",c++);
                }
                printf("\n");
        }
}
```



```
A B C D E
A B C D
A B C
A B
A
```

Experiment – 8

Write given programs for concepts of array.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

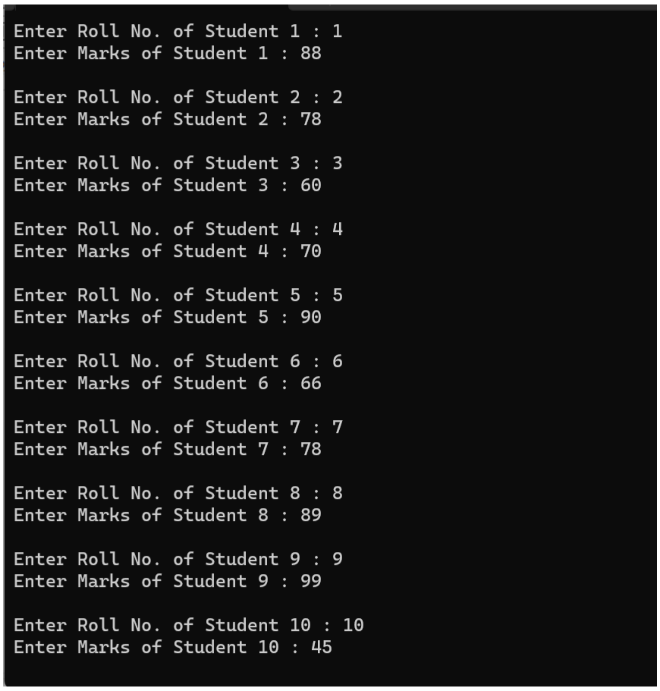
Objectives: (a) To understand the requirement of array to solve the given problem.

(b) To understand the syntax and concept of array in C Program.

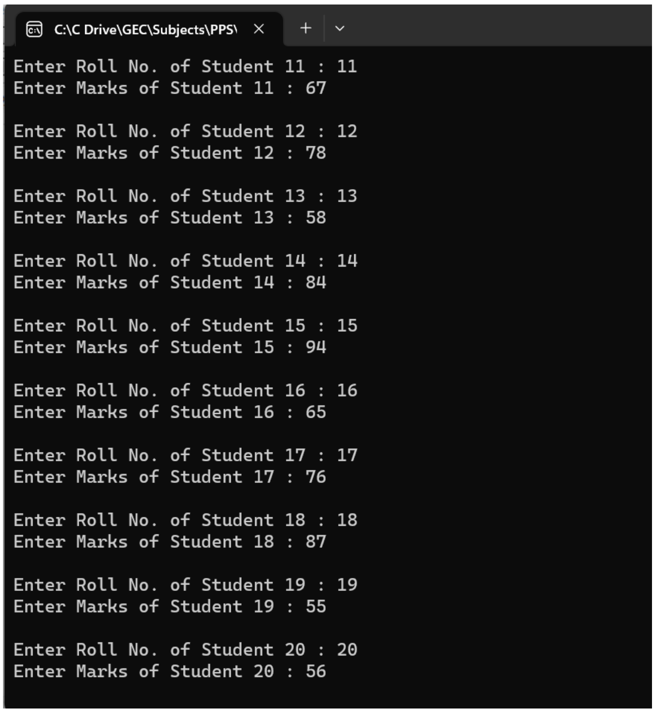
Equipment/Instruments: Computer with C Compiler

1) Write a C program to read and store the roll no and marks of 20 students using array.

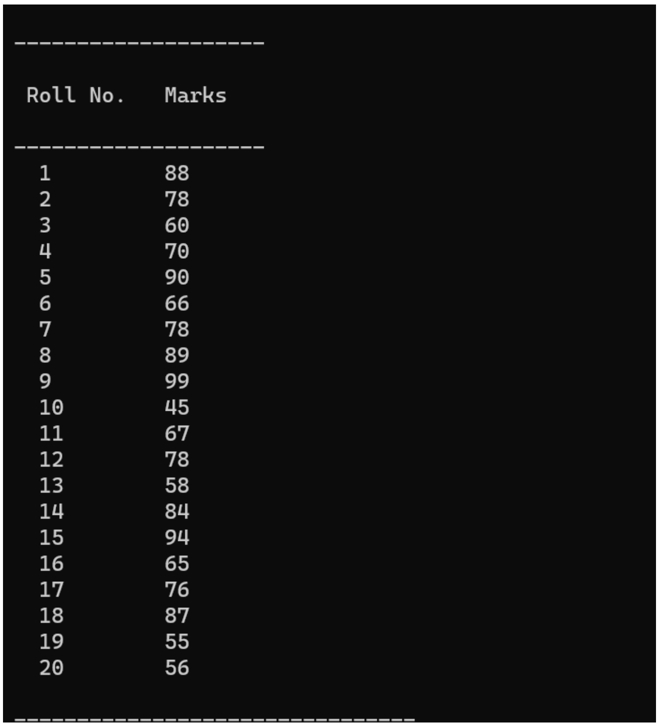
```
#include<stdio.h>
void main()
{
	int A[20][2];
	int i,j;

	for(i=0;i<20;i++)
	{
		printf("Enter Roll No. of Student %d : ", i+1);
		scanf("%d",&A[i][0]);

		printf("Enter Marks of Student %d : ", i+1);
		scanf("%d",&A[i][1]);

		printf("\n");
	}

	printf("\n--------------------\n");

	printf("\n Roll No.   Marks   \n");

	printf("\n--------------------\n");

	for(i=0;i<20;i++)
	{
		printf("  %d \t    %d \n",A[i][0],A[i][1]);
	}
}
```



```
Enter Roll No. of Student 1 : 1
Enter Marks of Student 1 : 88

Enter Roll No. of Student 2 : 2
Enter Marks of Student 2 : 78

Enter Roll No. of Student 3 : 3
Enter Marks of Student 3 : 60

Enter Roll No. of Student 4 : 4
Enter Marks of Student 4 : 70

Enter Roll No. of Student 5 : 5
Enter Marks of Student 5 : 90

Enter Roll No. of Student 6 : 6
Enter Marks of Student 6 : 66

Enter Roll No. of Student 7 : 7
Enter Marks of Student 7 : 78

Enter Roll No. of Student 8 : 8
Enter Marks of Student 8 : 89

Enter Roll No. of Student 9 : 9
Enter Marks of Student 9 : 99

Enter Roll No. of Student 10 : 10
Enter Marks of Student 10 : 45
```



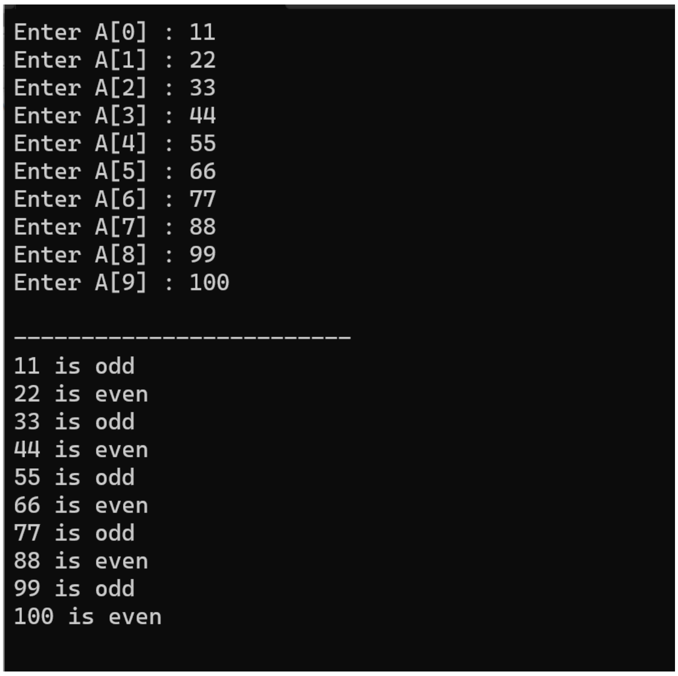
```
Enter Roll No. of Student 11 : 11
Enter Marks of Student 11 : 67

Enter Roll No. of Student 12 : 12
Enter Marks of Student 12 : 78

Enter Roll No. of Student 13 : 13
Enter Marks of Student 13 : 58

Enter Roll No. of Student 14 : 14
Enter Marks of Student 14 : 84

Enter Roll No. of Student 15 : 15
Enter Marks of Student 15 : 94

Enter Roll No. of Student 16 : 16
Enter Marks of Student 16 : 65

Enter Roll No. of Student 17 : 17
Enter Marks of Student 17 : 76

Enter Roll No. of Student 18 : 18
Enter Marks of Student 18 : 87

Enter Roll No. of Student 19 : 19
Enter Marks of Student 19 : 55

Enter Roll No. of Student 20 : 20
Enter Marks of Student 20 : 56
```



```
------------------------

 Roll No.     Marks

------------------------
  1           88
  2           78
  3           60
  4           70
  5           90
  6           66
  7           78
  8           89
  9           99
  10          45
  11          67
  12          78
  13          58
  14          84
  15          94
  16          65
  17          76
  18          87
  19          55
  20          56

--------------------------------------
```

2) Write a program to find out which number is even or odd from list of 10 numbers using array.

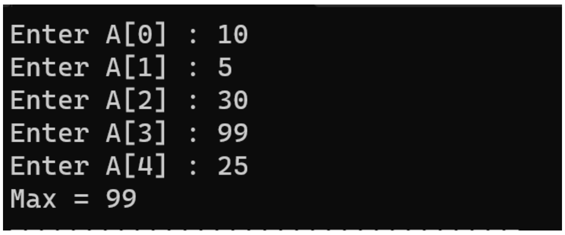
```
#include<stdio.h>
void main()
{
        int A[10];

        int i;

        for(i=0; i<10 ; i++)
        {
                printf("Enter A[%d] : ",i);
                scanf("%d",&A[i]);
        }

        printf("\n-------------------------\n");

        for(i=0; i<10 ; i++)
        {
                if(A[i]%2 == 0)
                {
                        printf("%d is even \n",A[i]);
                }
                else
                {
                        printf("%d is odd \n",A[i]);
                }
        }
}
```



```
Enter A[0] : 11
Enter A[1] : 22
Enter A[2] : 33
Enter A[3] : 44
Enter A[4] : 55
Enter A[5] : 66
Enter A[6] : 77
Enter A[7] : 88
Enter A[8] : 99
Enter A[9] : 100

-------------------------
11 is odd
22 is even
33 is odd
44 is even
55 is odd
66 is even
77 is odd
88 is even
99 is odd
100 is even
```

3) Write a program to find maximum element from 1-Dimensional array.

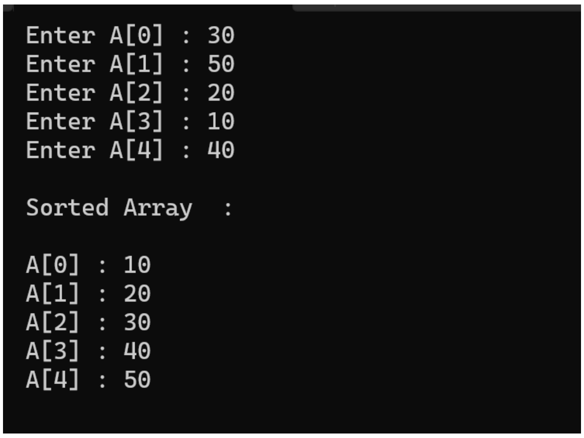
```
#include<stdio.h>
void main()
{
        int A[5];
        int i,max;

        for(i=0 ; i<5 ; i++)
        {
                printf("Enter A[%d] : ", i);
                scanf("%d",&A[i]);

                if(i==0)
                {
                        max = A[i];
                }
                else
                {
                        if(max < A[i])
                        {
                                max = A[i];
                        }
                }
        }
        printf("Max = %d",max);
}
```



```
Enter A[0] : 10
Enter A[1] : 5
Enter A[2] : 30
Enter A[3] : 99
Enter A[4] : 25
Max = 99
```

4) Write a program to sort given array in ascending order.

```
#include<stdio.h>

void main()
{
	int A[5];
	int i,j,temp;

	for(i=0; i<5 ;i++)
	{
		printf(" Enter A[%d] : ",i);
		scanf("%d",&A[i]);
	}

	for(i=0; i<4 ; i++)
	{
		for(j=i+1 ; j<5 ; j++)
		{
			if(A[i]>A[j])
			{
				temp = A[i];
				A[i] = A[j];
				A[j] = temp;
			}
		}
	}

	printf("\n Sorted Array  : \n\n");
	for(i=0; i<5 ;i++)
	{
		printf(" A[%d] : %d \n",i,A[i]);
	}

}
```



```
 Enter A[0] : 30
 Enter A[1] : 50
 Enter A[2] : 20
 Enter A[3] : 10
 Enter A[4] : 40

 Sorted Array  :

 A[0] : 10
 A[1] : 20
 A[2] : 30
 A[3] : 40
 A[4] : 50
```

Experiment – 9

Write given programs for string manipulation.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

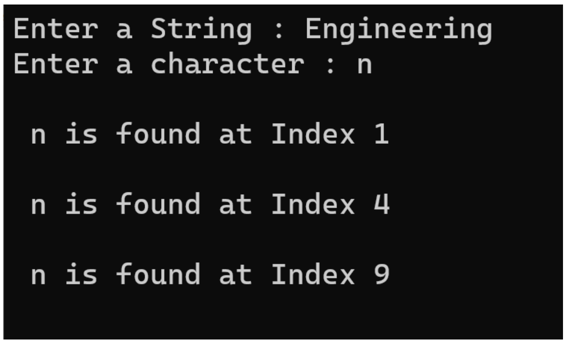
Objectives: (a) To understand the declaration and use of string.

(b) To understand inbuilt string functions in C.

Equipment/Instruments: Computer with C Compiler

1) Write a program to find a character from given string.

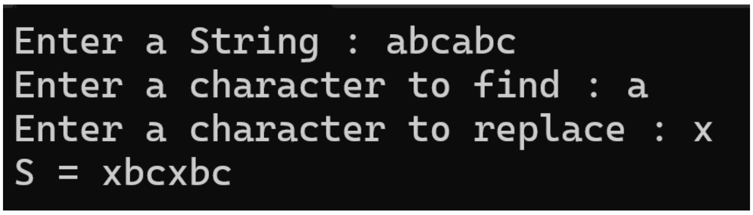
```
#include<stdio.h>
#include<string.h>

void main()
{
        char S[20];
        char c;
        int i,len,flag=0;

        printf("Enter a String : ");
        gets(S);

        printf("Enter a character : ");
        c=getchar();

        len=strlen(S);

        for(i=0 ; i<len ; i++)
        {
                if(S[i] == c)
                {
                        printf("\n %c is found at Index %d \n", c , i);
                        flag=1;
                }
        }

        if(flag==0)
        {
                printf("%c is not found in the given string",c);
        }
}
```



```
Enter a String : Engineering
Enter a character : n

 n is found at Index 1

 n is found at Index 4

 n is found at Index 9
```

2) Write a program to replace a character in given string.

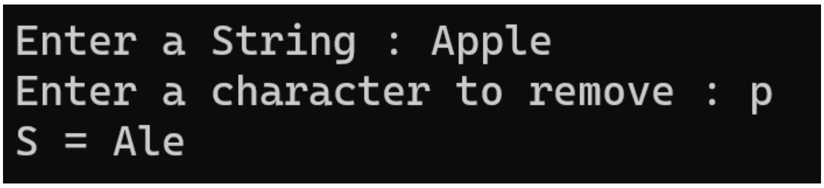
```
#include<stdio.h>
#include<string.h>

void main()
{
	char S[20];
	char c,d;
	int i,len,flag=0;

	printf("Enter a String : ");
	gets(S);

	printf("Enter a character to find : ");
	c=getchar();

	fflush(stdin);

	printf("Enter a character to replace : ");
	d=getchar();

	len=strlen(S);

	for(i=0 ; i<len ; i++)
	{
		if(S[i] == c)
		{
			S[i] = d;
		}
	}
	printf("S = %s",S);
}
```



```
Enter a String : abcabc
Enter a character to find : a
Enter a character to replace : x
S = xbcxbc
```

3) Write a program to delete a character in given string.

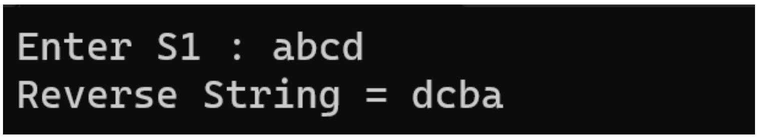
```
#include<stdio.h>
#include<string.h>

void main()
{
	char S[20];
	char c;
	int i,j,len;

	printf("Enter a String : ");
	gets(S);

	printf("Enter a character to remove : ");
	c=getchar();

	len=strlen(S);

	for(i=0 ; i<len ; i++)
	{
		if(S[i] == c)
		{
			for(j=i ; j<len ; j++)
			{
				S[j] = S[j+1];
			}

			i--;
		}
	}
	printf("S = %s",S);
}
```



```
Enter a String : Apple
Enter a character to remove : p
S = Ale
```

4) Write a program to reverse string.

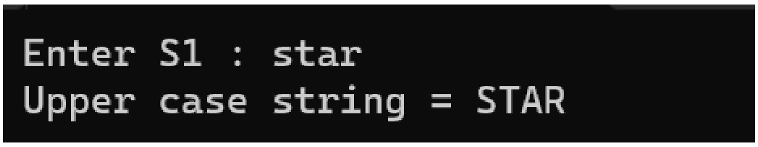
```
#include<stdio.h>
#include<string.h>

void main()
{
        char S1[10];

        printf("Enter S1 : ");
        gets(S1);s

        strrev(S1);

        printf("Reverse String = %s", S1);
}
```



```
Enter S1 : abcd
Reverse String = dcba
```

5) Write a program to convert string into upper case.

```
#include<stdio.h>
#include<string.h>

void main()
{
        char S1[10];

        printf("Enter S1 : ");
        gets(S1);

        strupr(S1);

        printf("Upper case string = %s ",S1);
}
```



```
Enter S1 : star
Upper case string = STAR
```

Experiment – 10

Write given programs using User Defined Functions.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

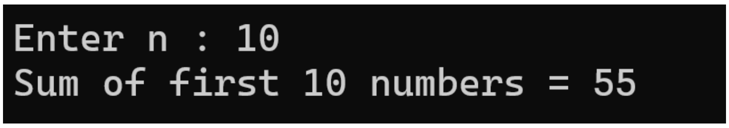
Objectives: (a) To understand the need of user defined functions to solve the given problem.
(b) To understand function declaration, function definition and function call.

Equipment/Instruments: Computer with C Compiler

.

1) Write a program that defines a function to add first n numbers.

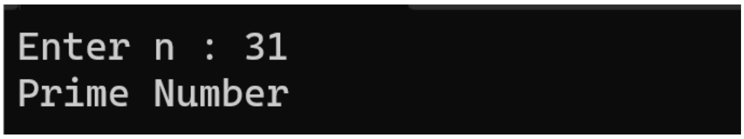
```
#include<stdio.h>

void Add(int);
void main()
{
        int n;

        printf("Enter n : ");
        scanf("%d",&n);

        Add(n);
}
void Add(int n)
{
        int I, sum=0;

        for(i=1; i<=n ; i++)
        {
                sum = sum + I;
        }

        printf("Sum of first %d numbers = %d", n,sum);
}
```



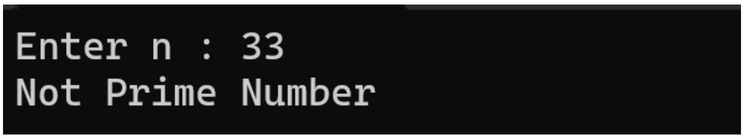
```
Enter n : 10
Sum of first 10 numbers = 55
```

2) Write a function in the program to return 1 if number is prime otherwise return 0.

```
#include<stdio.h>

int IsPrime(int n);
void main()
{
        int n,flag,I;

        printf(“Enter n : “);
        scanf(“%d”,&n);

        flag=IsPrime(n);

        if(flag == 1)
        {
                printf(“Prime Number”);
        }
        else
        {
                printf(“Not Prime Number”);
        }
}
int IsPrime(int n)
{
        int I,flag=1;

        for(i=2 ; i<n; i++)
        {
                if(n%i == 0)
                {
                        flag=0;
                        break;
                }
        }
        return flag;
}
```



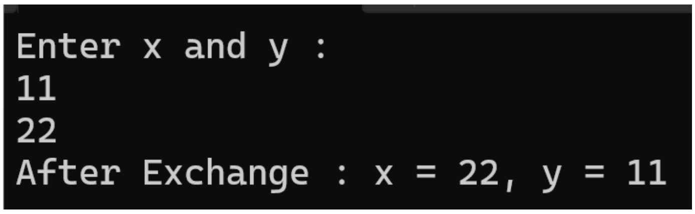
```
Enter n : 31
Prime Number
```



```
Enter n : 33
Not Prime Number
```

3) Write a function Exchange to interchange the values of two variables, say x and y using a function.

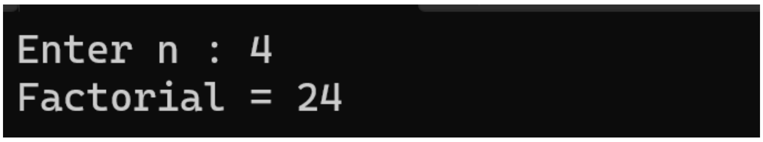
```
#include<stdio.h>
void Exchange(int*,int*);
void main()
{
        int x,y;

        printf("Enter x and y : \n");
        scanf("%d%d",&x,&y);

        Exchange(&x,&y);

        printf("After Exchange : x = %d, y = %d ",x,y);

}
void Exchange(int *a,int *b)
{
        int Temp ;

        Temp = *a;
        *a = *b;
        *b = Temp;
}
```



```
Enter x and y :
11
22
After Exchange : x = 22, y = 11
```

4) Write a program to find factorial of a number using recursion.

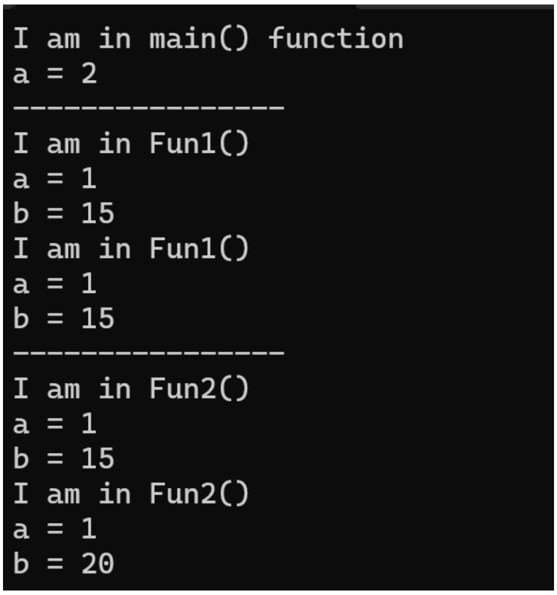
```
#include<stdio.h>

int Factorial(int);
void main()
{
	int n, ans;

	printf(“Enter n : “);
	scanf(“%d”,&n);

	ans = Factorial(n);

	printf(“Factorial = %d”,ans);
}
int Factorial(int n)
{
	if(n==0 || n==1)
	{
		return 1;
	}
	else
	{
		return n * Factorial(n-1);
	}
}
```



```
Enter n : 4
Factorial = 24
```

5) Write a C program using global variable, static variable.

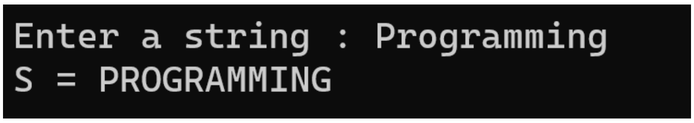
```
#include<stdio.h>
int a=1;
void main()
{
        int a=2;

        printf(“I am in main() function \n”);
        printf(“a = %d \n”,a);
        printf(“----------------\n”);
        Fun1();
        Fun1();
        printf(“----------------\n”);
        Fun2();
        Fun2();
}
void Fun1()
{
        int b=10;

        printf(“I am in Fun1() \n”);
        printf(“a = %d \n”,a);
        b = b+5;
        printf(“b = %d \n”,b);
}
void Fun2()
{
        static int b=10;

        printf(“I am in Fun2() \n”);
        printf(“a = %d \n”,a);
        b = b+5;
        printf(“b = %d \n”,b);
}
```



```
I am in main() function
a = 2
----------------
I am in Fun1()
a = 1
b = 15
I am in Fun1()
a = 1
b = 15
----------------
I am in Fun2()
a = 1
b = 15
I am in Fun2()
a = 1
b = 20
```

6) Write a function that will scan a character string passed as an argument and convert all lowercase character into their uppercase equivalents

```
#include<stdio.h>
#include<ctype.h>

void ToUpper(char S[])
{
        int I, len;
        len = strlen(S);

        for(i=0 ; i<len ; i++)
        {
                if( islower(S[i]) )
                {
                        S[i] = toupper(S[i]);
                }
        }
}
void main()
{
        char S[20];

        printf(“Enter a string : “);
        gets(S);

        ToUpper(S);             // &S[0] = S
        printf(“S = %s”,S);
}
```



```
Enter a string : Programming
S = PROGRAMMING
```

Experiment – 11

Write given programs using concepts of Structure.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

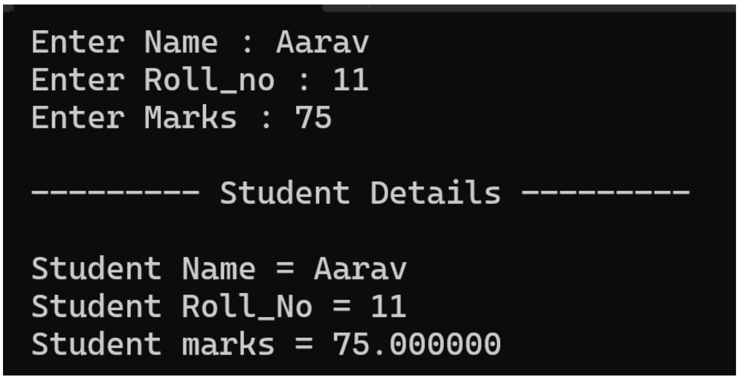
Objectives: (a) To understand the syntax and concepts of structure in C Program.

(b) To understand user defined datatypes.

Equipment/Instruments: Computer with C Compiler

1) Write a program to read structure elements from keyboard.

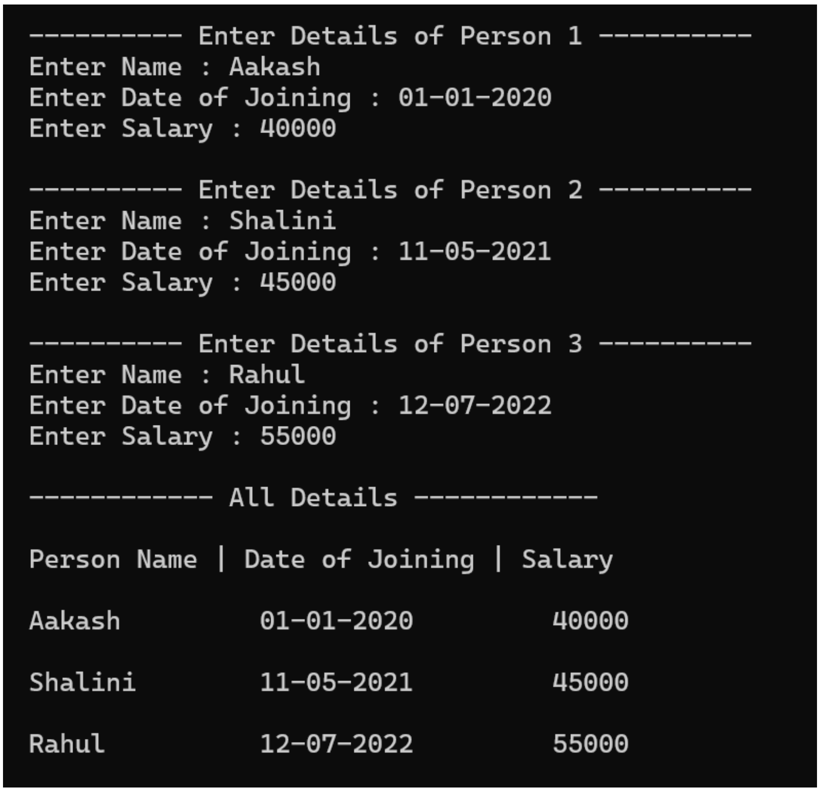
```
#include<stdio.h>

struct Student
{
	char Name[20];
	int Roll_no;
	float Marks;
}S1;

void main()
{
	printf(" Enter Name : ");
	scanf("%s",S1.Name);

	printf(" Enter Roll_no : ");
	scanf("%d",&S1.Roll_no );

	printf(" Enter Marks : ");
	scanf("%f", &S1.Marks );

	printf("\n --------- Student Details --------- \n");

	printf("\n Student Name = %s ",S1.Name);
	printf("\n Student Roll_No = %d ",S1.Roll_no);
	printf("\n Student marks = %f ",S1.Marks);
}
```



```
Enter Name : Aarav
Enter Roll_no : 11
Enter Marks : 75

--------- Student Details ---------

Student Name = Aarav
Student Roll_No = 11
Student marks = 75.000000
```

2) Define a structure type struct personal that would contain person name, date of joining and salary using this structure to read this information of 5 people and print the same on screen.

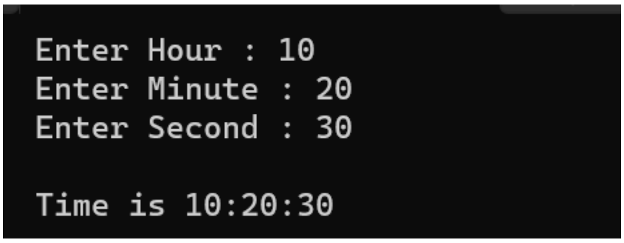
```
#include<stdio.h>

struct Person
{
	char Name[20];
	char DOJ[10];
	int Salary;

}P[3];

void main()
{
	int i;

	for(i=0;i<3;i++)
	{
		printf("\n ---------- Enter Details of Person %d ---------- \n",i+1);

		printf(" Enter Name : ");
		scanf("%s",P[i].Name);

		printf(" Enter Date of Joining : ");
		scanf("%s",P[i].DOJ );

		printf(" Enter Salary : ");
		scanf("%d",&P[i].Salary);
	}

	printf("\n ------------ All Details ------------ \n");
	printf("\n Person Name | Date of Joining | Salary \n");

	for(i=0;i<3;i++)
	{
		printf("\n %-14s %-17s  %-7d  \n",P[i].Name,P[i].DOJ,P[i].Salary);
	}
}
```



```
---------- Enter Details of Person 1 ----------
Enter Name : Aakash
Enter Date of Joining : 01-01-2020
Enter Salary : 40000

---------- Enter Details of Person 2 ----------
Enter Name : Shalini
Enter Date of Joining : 11-05-2021
Enter Salary : 45000

---------- Enter Details of Person 3 ----------
Enter Name : Rahul
Enter Date of Joining : 12-07-2022
Enter Salary : 55000

------------ All Details ------------

Person Name | Date of Joining | Salary

Aakash         01-01-2020         40000

Shalini        11-05-2021         45000

Rahul          12-07-2022         55000
```

3) Define structure data type called time_struct containing three member's integer hour, integer minute and integer second. Develop a program that would assign values to the individual number and display the time in the following format: 16: 40:51

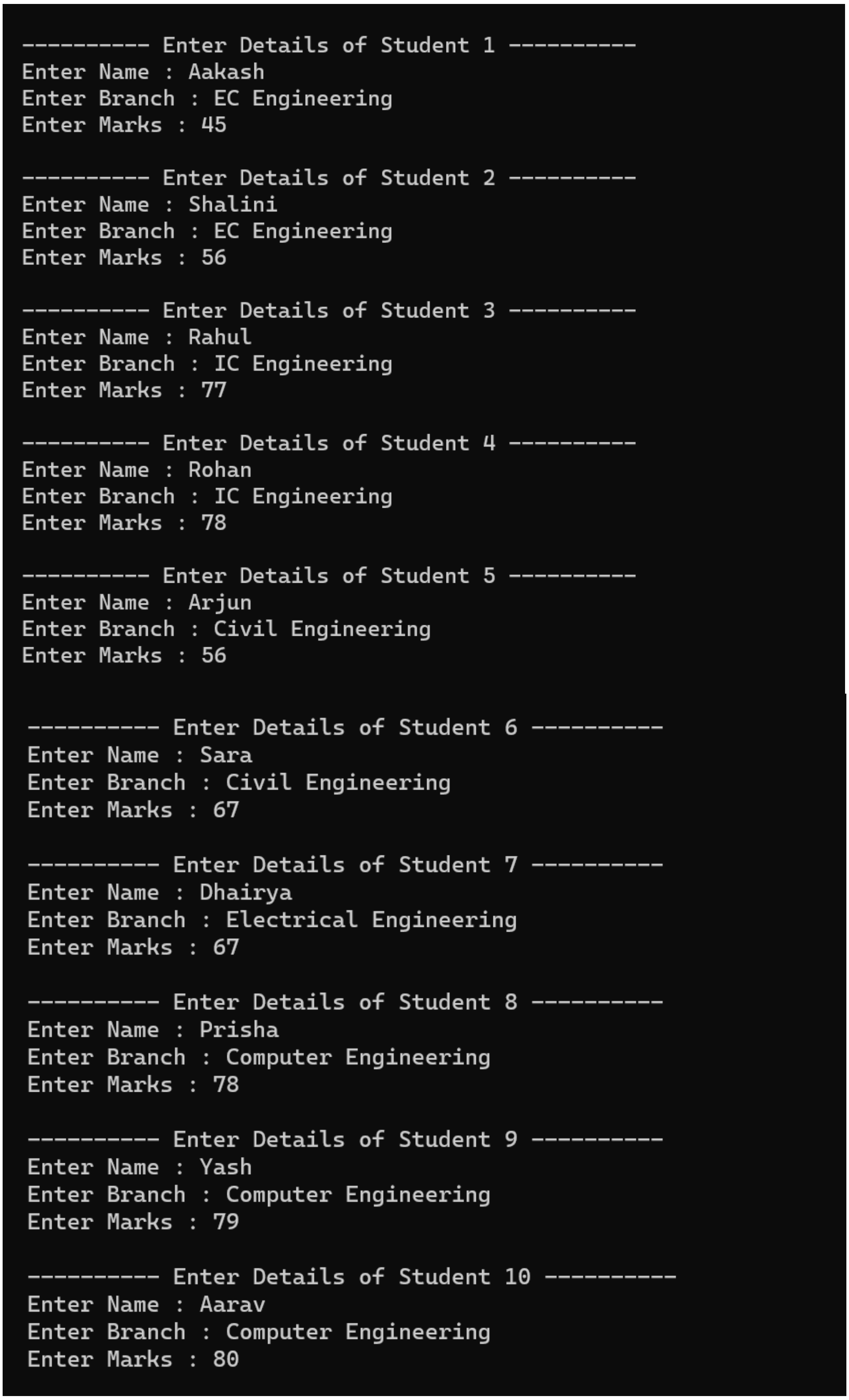
```
#include<stdio.h>

struct Time
{
        int hour;
        int min;
        int sec;

}T1;

void Display(struct Time);

void main()
{
        printf(" Enter Hour : ");
        scanf("%d",&T1.hour);

        printf(" Enter Minute : ");
        scanf("%d",&T1.min);

        printf(" Enter Second : ");
        scanf("%d",&T1.sec);

        Display(T1);
}

void Display(struct Time T1)
{
        printf("\n Time is %d:%d:%d",T1.hour,T1.min,T1.sec);
}
```



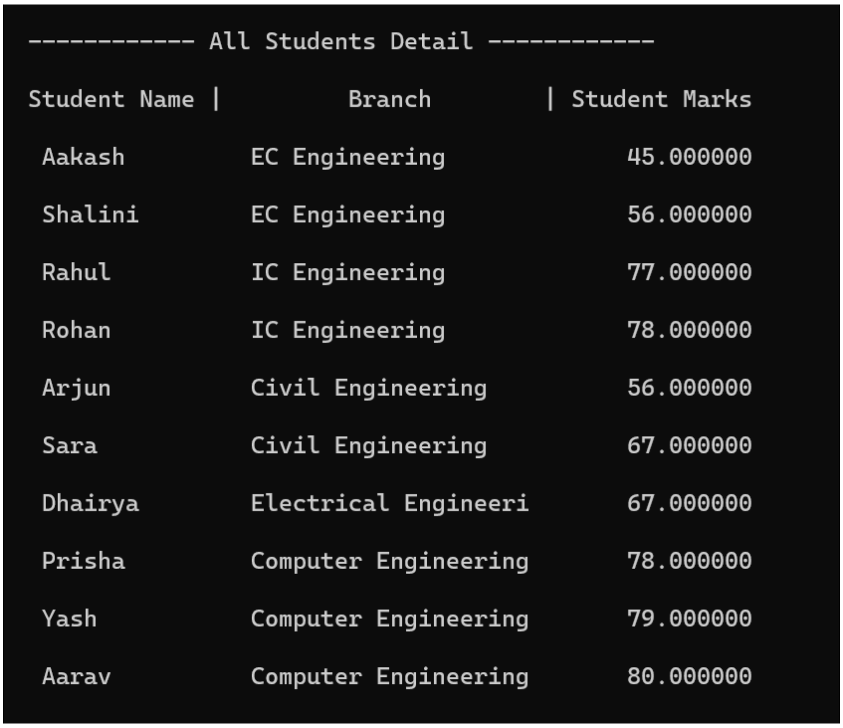
```
 Enter Hour : 10
 Enter Minute : 20
 Enter Second : 30

 Time is 10:20:30
```

4) Design a structure student_record to contain name, branch and total marks obtained. Develop a program to read data for 10 students in a class and print them.

```
#include<stdio.h>

struct student_record
{
        char Name[20];
        char Branch[20];
        float Marks;

}S[10];

void main()
{
        int i;

        for(i=0;i<10;i++)
        {
                printf("\n ---------- Enter Details of Student %d ---------- \n",i+1);

                printf(" Enter Name : ");
                gets(S[i].Name);

                fflush(stdin);

                printf(" Enter Branch : ");
                gets(S[i].Branch );

                fflush(stdin);

                printf(" Enter Marks : ");
                scanf("%f",&S[i].Marks );

                fflush(stdin);

        }
        printf("\n ------------ All Students Detail ------------ \n");
        printf("\n Student Name |       Branch      | Student Marks \n");
        for(i=0;i<10;i++)
        {
                printf("\n  %s  \t %s \t    %f  \n",S[i].Name,S[i].Branch,S[i].Marks);
        }
}
```



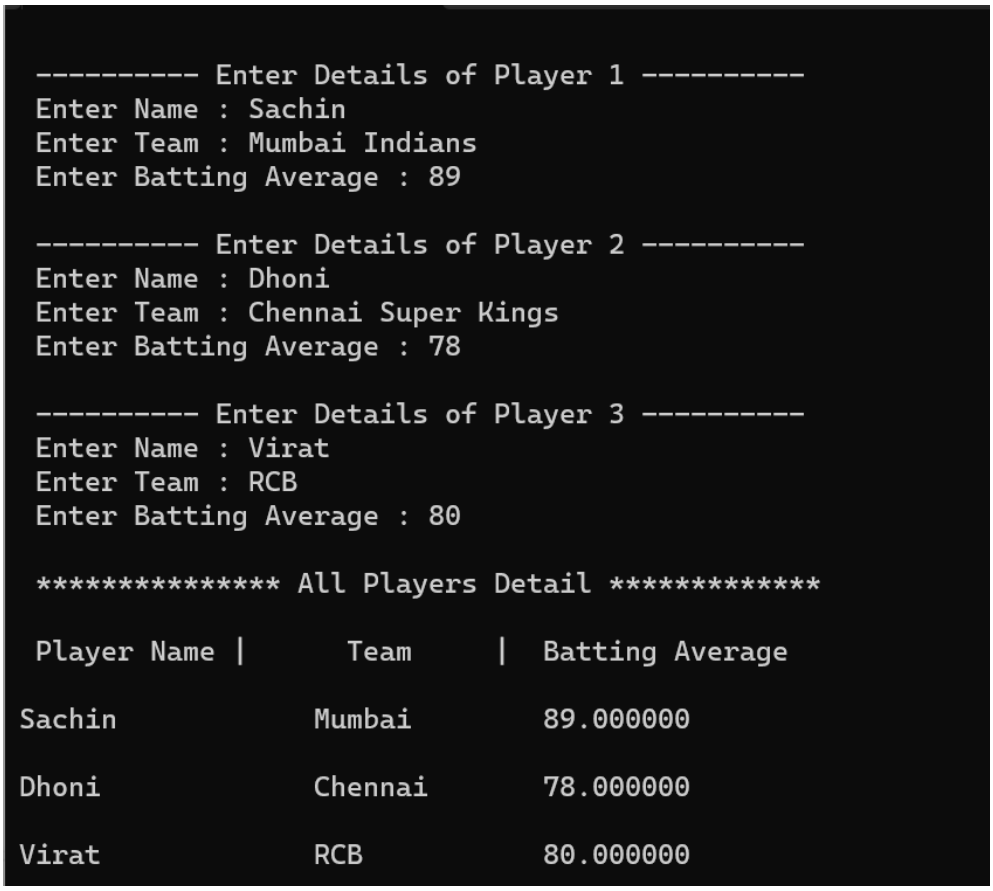
```
---------- Enter Details of Student 1 ----------
Enter Name : Aakash
Enter Branch : EC Engineering
Enter Marks : 45

---------- Enter Details of Student 2 ----------
Enter Name : Shalini
Enter Branch : EC Engineering
Enter Marks : 56

---------- Enter Details of Student 3 ----------
Enter Name : Rahul
Enter Branch : IC Engineering
Enter Marks : 77

---------- Enter Details of Student 4 ----------
Enter Name : Rohan
Enter Branch : IC Engineering
Enter Marks : 78

---------- Enter Details of Student 5 ----------
Enter Name : Arjun
Enter Branch : Civil Engineering
Enter Marks : 56

---------- Enter Details of Student 6 ----------
Enter Name : Sara
Enter Branch : Civil Engineering
Enter Marks : 67

---------- Enter Details of Student 7 ----------
Enter Name : Dhairya
Enter Branch : Electrical Engineering
Enter Marks : 67

---------- Enter Details of Student 8 ----------
Enter Name : Prisha
Enter Branch : Computer Engineering
Enter Marks : 78

---------- Enter Details of Student 9 ----------
Enter Name : Yash
Enter Branch : Computer Engineering
Enter Marks : 79

---------- Enter Details of Student 10 ----------
Enter Name : Aarav
Enter Branch : Computer Engineering
Enter Marks : 80
```



```
------------ All Students Detail ------------

Student Name |         Branch        | Student Marks

 Aakash         EC Engineering             45.000000

 Shalini        EC Engineering             56.000000

 Rahul          IC Engineering             77.000000

 Rohan          IC Engineering             78.000000

 Arjun          Civil Engineering          56.000000

 Sara           Civil Engineering          67.000000

 Dhairya        Electrical Engineeri       67.000000

 Prisha         Computer Engineering       78.000000

 Yash           Computer Engineering       79.000000

 Aarav          Computer Engineering       80.000000
```

5) Define a structure called cricket that will describe the following information:
Player name,
Team name,
Batting average,
Using cricket declare an array player with 50 elements and write a C program to read the information about all the 50 players and print list containing names of players with their batting average.

```
#include<stdio.h>

struct Player
{
        char Name[20];
        char Team[20];
        float Bat_avg;

}P[3];

void main()
{
        int i;

        for(i=0;i<3;i++)
        {
                printf("\n ---------- Enter Details of Player %d ---------- \n",i+1);

                printf(" Enter Name : ");
                scanf("%s",P[i].Name);

                printf(" Enter Team : ");
                scanf("%s",P[i].Team );

                printf(" Enter Batting Average : ");
                scanf("%f",&P[i].Bat_avg );

        }

        printf("\n *************** All Players Detail ************* \n");
        printf("\n Player Name |      Team     | Batting Average \n");
        for(i=0;i<3;i++)
        {
                printf("\n%-17s %-12s  %f \n",P[i].Name,P[i].Team,P[i].Bat_avg );
        }

}
```



```
 ---------- Enter Details of Player 1 ----------
 Enter Name : Sachin
 Enter Team : Mumbai Indians
 Enter Batting Average : 89

 ---------- Enter Details of Player 2 ----------
 Enter Name : Dhoni
 Enter Team : Chennai Super Kings
 Enter Batting Average : 78

 ---------- Enter Details of Player 3 ----------
 Enter Name : Virat
 Enter Team : RCB
 Enter Batting Average : 80

 *************** All Players Detail *************

 Player Name |      Team     |  Batting Average

Sachin            Mumbai        89.000000

Dhoni             Chennai       78.000000

Virat             RCB           80.000000
```

Experiment – 12

Write given programs using concepts of Pointers.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

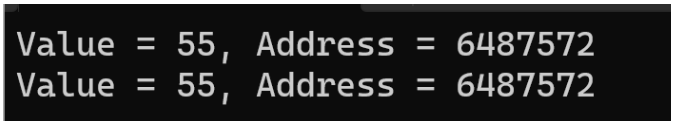
Objectives: (a) To understand the concepts of pointers in C Program.

(b) To understand the method to access variables and arrays using pointers.

Equipment/Instruments: Computer with C Compiler

1) Write a program to print address of variable using pointer.

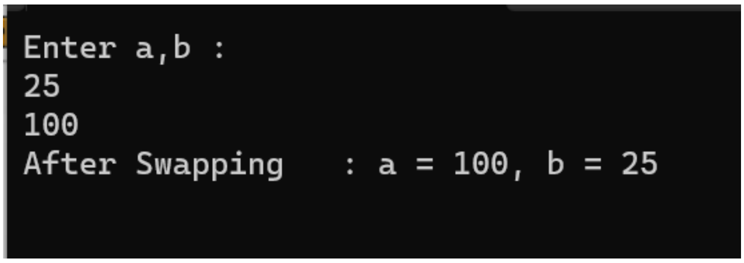
```
#include<stdio.h>

void main()
{
        int a, *p;

        a = 55;
        p = &a;

        printf("Value = %d, Address = %u \n",a, &a);
        printf("Value = %d, Address = %u",*p, p);

}
```



```
Value = 55, Address = 6487572
Value = 55, Address = 6487572
```

2) Write a C program to swap the two values using pointers.

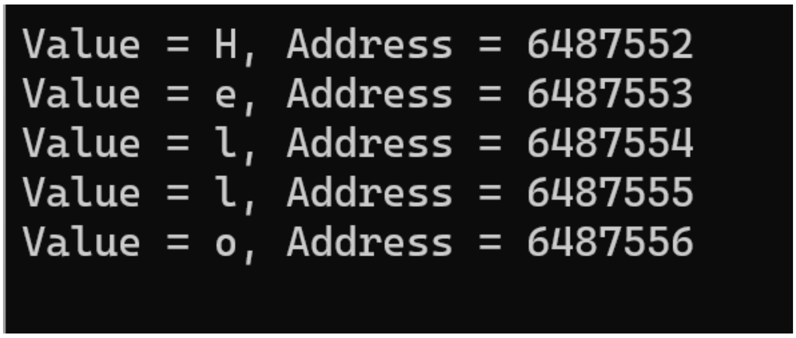
```
#include<stdio.h>

void swap(int*,int*);
void main()
{
        int a,b;

        printf("Enter a,b : \n");
        scanf("%d%d",&a,&b);

        swap(&a,&b);

        printf("After Swapping \t : a = %d, b = %d \n\n",a,b);
}
void swap(int *a,int *b)
{
        int Temp ;

        Temp = *a;
        *a = *b;
        *b = Temp;
}
```



```
Enter a,b :
25
100
After Swapping   : a = 100, b = 25
```

3) Write a C program to print the address of character and the character of string using a pointer.

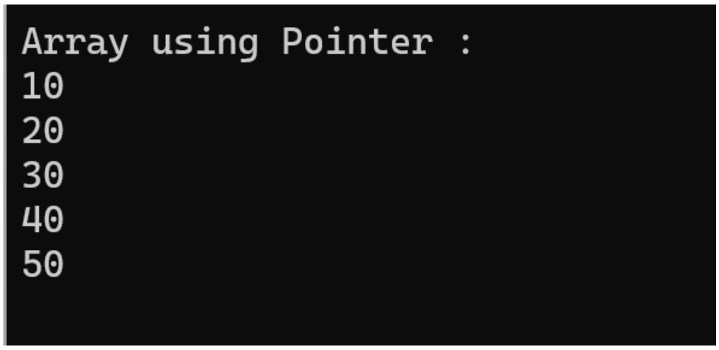
```
#include<stdio.h>
#include<string.h>

void main()
{
        char str[10]="Hello";
        char *p;
        int i,len;

        len=strlen(str);
        p=str;

        for(i=0;i<len;i++)
        {
                printf("Value = %c, Address = %u \n",*p,p);
                p++;
        }
}
```



```
Value = H, Address = 6487552
Value = e, Address = 6487553
Value = l, Address = 6487554
Value = l, Address = 6487555
Value = o, Address = 6487556
```

4) Write a program to access elements using pointer.

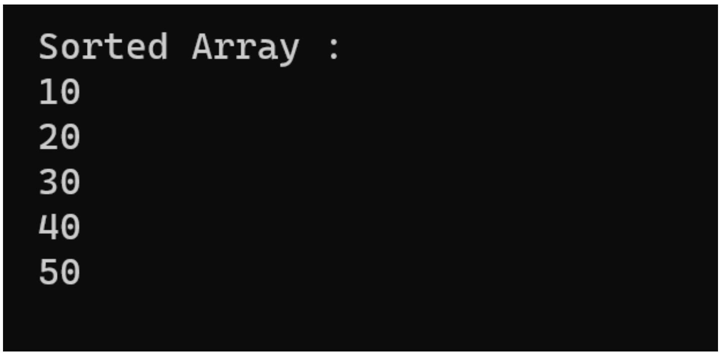
```
#include<stdio.h>
void main()
{
        int A[5] = {10,20,30,40,50};

        int i,*p;

        p=A;   // p = &A[0];

        printf("Array using Pointer : \n");

        for(i=0 ; i<5 ; i++)
        {
                printf("%d \n", *(p+i));
        }
}
```



```
Array using Pointer :
10
20
30
40
50
```

5) Write a program for sorting using pointer.

```
#include<stdio.h>
void main()
{
        int A[5] = {30,50,20,40,10};

        int *p;

        int i,j,temp;

        p = A;

        for(i=0 ; i<=3 ; i++)
        {
                for(j=i+1 ; j<=4 ; j++)
                {
                        if(*(p+i) > *(p+j))
                        {
                                temp = *(p+i);
                                *(p+i) = *(p+j);
                                *(p+j) = temp;
                        }
                }
        }

        printf("\n Sorted Array : \n");

        for(i=0 ; i<=4 ; i++)
        {
                printf(" %d \n", *(p+i));
        }
}
```



```
 Sorted Array :
 10
 20
 30
 40
 50
```

Experiment – 13

Write given programs using concepts of File handling.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

Relevant CO: CO-2

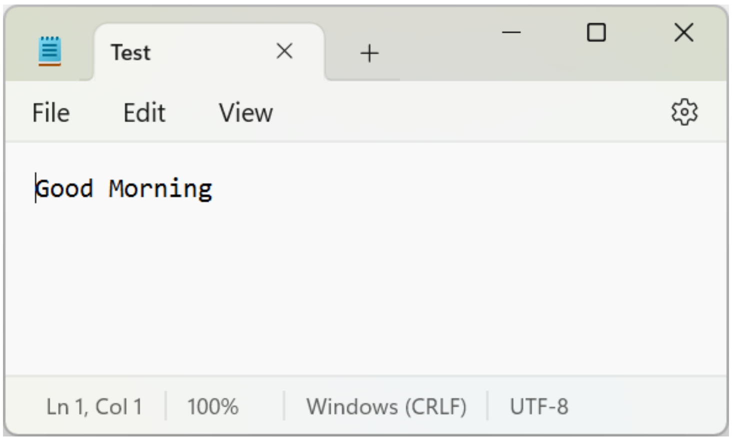
Objectives: (a) To understand the file handling functions C Program.

(b) To read and write files using C program.

Equipment/Instruments: Computer with C Compiler

1) Write a program to write a string in file.

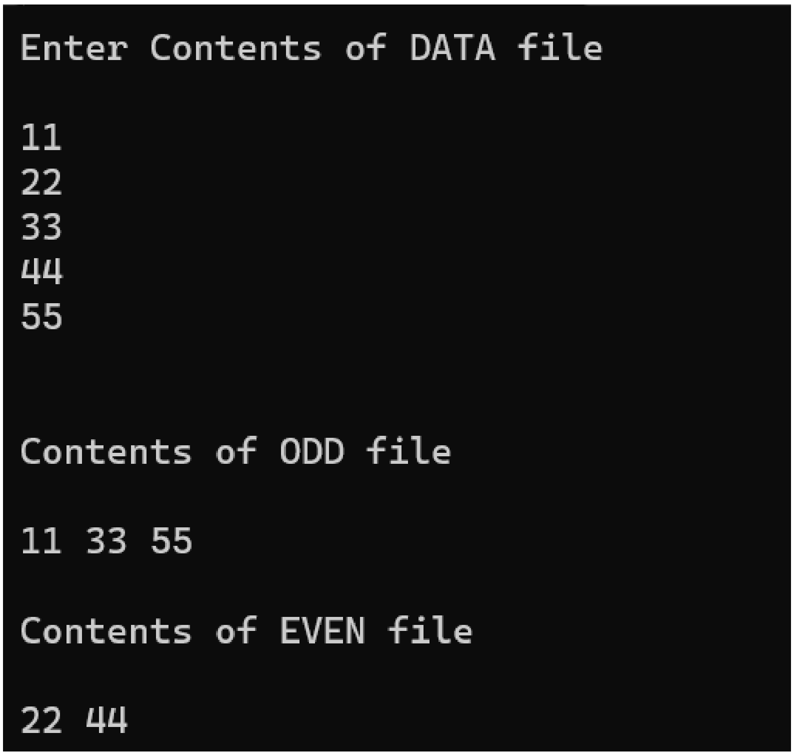
```
#include<stdio.h>

void main()
{
        FILE *fp;
        char *str="Good Morning";

        fp = fopen("D:\Test.txt","w");

        fputs(str,fp);

        printf("String written successfully in file.");
}
```



```
Good Morning
```

2) A file named data contains series of integer numbers. Write a c program to read all numbers from file and then write all odd numbers into file named “odd” and write all even numbers into file named “even”. Display all the contents of these file on screen.

```
#include <stdio.h>
main()
{
        FILE *f1, *f2, *f3;
        int a, i;
        printf("Enter Contents of DATA file\n\n");

        f1 = fopen("DATA.txt", "w");                /* Create DATA file */
        for(i = 1; i <= 5; i++)
        {
            scanf("%d", &a);
            putw(a,f1);
        }
        fclose(f1);

        f1 = fopen("DATA.txt", "r");
        f2 = fopen("ODD.txt", "w");
        f3 = fopen("EVEN.txt", "w");

        while((a = getw(f1)) != EOF)                /* Read from DATA file */
        {
                if(a%2 == 0)
                        putw(a, f3);
                else
                        putw(a, f2);
        }
        fclose(f1);
        fclose(f2);
        fclose(f3);

        f2 = fopen("ODD.txt","r");
        f3 = fopen("EVEN.txt", "r");

        printf("\n\nContents of ODD file\n\n");

        while((a = getw(f2)) != EOF)
        {
                printf("%d ", a);
        }

        printf("\n\nContents of EVEN file\n\n");

        while((a = getw(f3)) != EOF)
        {
                printf("%d ", a);
        }
        fclose(f2);
        fclose(f3);
}
```



```
Enter Contents of DATA file

11
22
33
44
55


Contents of ODD file

11 33 55

Contents of EVEN file

22 44
```

Experiment – 14

Write given programs to generate mathematical series using control statements and functions.

Date:

Competency and Practical Skills: C basic programming syntax and datatypes

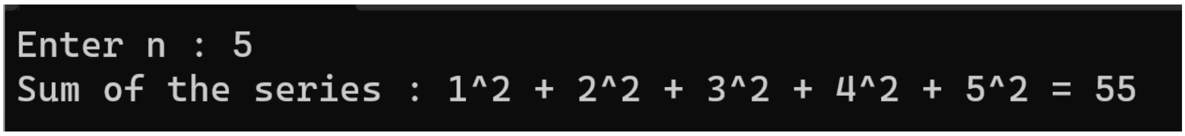
Relevant CO: CO-2

Objective: (a) To solve mathematical series using control statements and functions C Program.

Equipment/Instruments: Computer with C Compiler

1) Write a program to evaluate the series 1^2+2^2+3^2+……+n^2

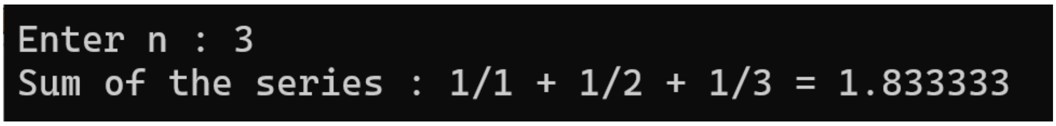
```
#include<stdio.h>
#include<math.h>

void main()
{
        int n,i;
        int sum=0;

        printf("Enter n : ");
        scanf("%d",&n);

        printf("Sum of the series : ");

        for (i=1 ; i<=n ; i++)
        {
                sum = sum + pow(i,2);

                if (i != n)
                {
                        printf("%d^2 + ",i);
                }
                else
                {
                        printf("%d^2 = %d ",i,sum);
                }
        }
}
```



```
Enter n : 5
Sum of the series : 1^2 + 2^2 + 3^2 + 4^2 + 5^2 = 55
```

2) Write a C program to find 1+1/2+1/3+1/4+. +1/n.

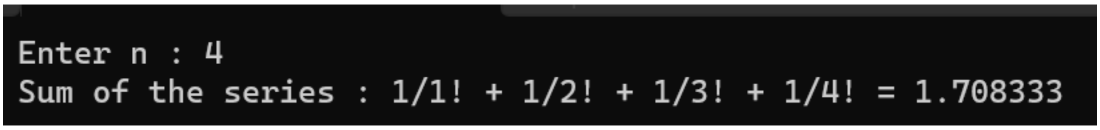
```
#include<stdio.h>
#include<math.h>
void main()
{
        int n,i;
        float sum=0.0;
        printf("Enter n : ");
        scanf("%d",&n);

        printf("Sum of the series : ");

        for (i=1 ; i<=n ; i++)
        {
                sum = sum + 1.0/i;

                if (i != n)
                {
                        printf("1/%d + ",i);
                }
                else
                {
                        printf("1/%d = %f ",i,sum);
                }
        }
}
```



```
Enter n : 3
Sum of the series : 1/1 + 1/2 + 1/3 = 1.833333
```

3) Write a C program to find 1+1/2!+1/3!+1/4!+…+1/n!.

```
#include<stdio.h>
#include<math.h>
int Fact(int n)
{
        int i,fact=1;

        for(i=1;i<=n;i++)
        {
                fact = fact * i;
        }
        return fact;
}
void main()
{
        int n,i;
        float sum=0;
        printf("Enter n : ");
        scanf("%d",&n);

        printf("Sum of the series : ");

        for (i=1 ; i<=n ; i++)
        {
                sum = sum + 1.0/Fact(i);

                if (i != n)
                {
                        printf("1/%d! + ",i);
                }
                else
                {
                        printf("1/%d! = %f ",i,sum);
                }
        }
}
```

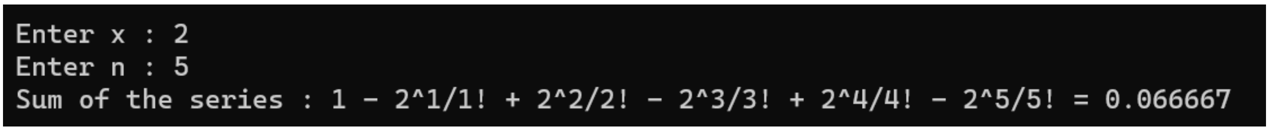


```
Enter n : 4
Sum of the series : 1/1! + 1/2! + 1/3! + 1/4! = 1.708333
```

4) Write a program to evaluate the series sum=1-x+x^2/2!-x^3/3!+x^4/4! ...-x^9/9!

```
#include<stdio.h>
#include<math.h>
int Fact(int n)
{
	int i,fact=1;
	for(i=1;i<=n;i++)
	{
		fact = fact * i;
	}
	return fact;
}
void main()
{
	int x,n,i;
	float sum=1.0;

	printf("Enter x : ");
	scanf("%d",&x);

	printf("Enter n : ");
	scanf("%d",&n);

	printf("Sum of the series : 1 - ");

	for (i=1 ; i<=n ; i++)
	{
		if(i%2 == 0)
		{
			sum = sum + pow(x,i)/Fact(i);
		}
		else
		{
			sum = sum - pow(x,i)/Fact(i);
		}
		//  ----------  print Series  --------
		if (i != n)
		{
			if(i%2 == 0)
			{
				printf("%d^%d/%d! - ",x,i,i);
			}
			else
			{
				printf("%d^%d/%d! + ",x,i,i);
			}
		}
		else
		{
			printf("%d^%d/%d! = %f",x,i,i,sum);
		}
```

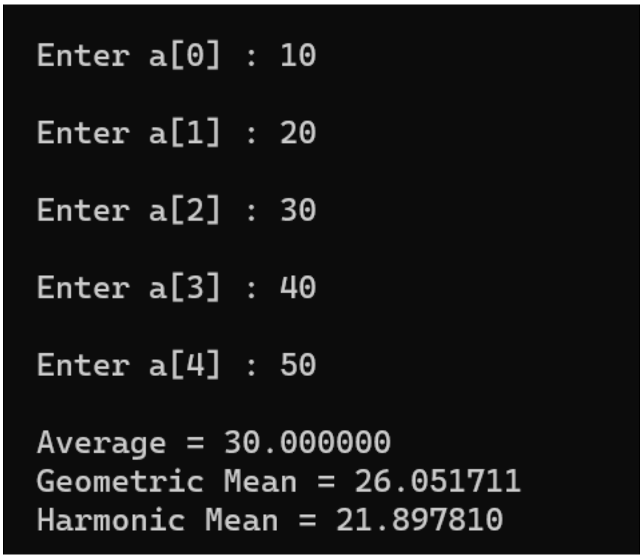
```
        }

}
```



```
Enter x : 2
Enter n : 5
Sum of the series : 1 - 2^1/1! + 2^2/2! - 2^3/3! + 2^4/4! - 2^5/5! = 0.066667
```

5) Write a C program to calculate the average, geometric and harmonic mean of n elements in an array.

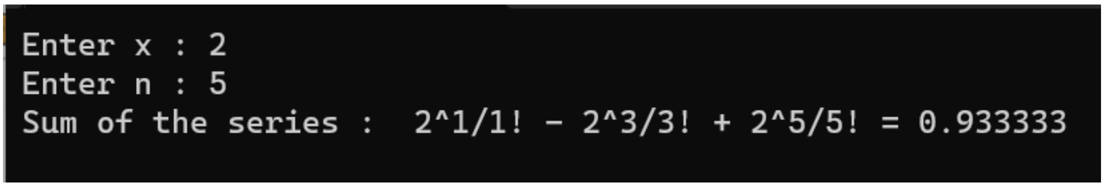
```
#include<stdio.h>
#include<math.h>

int main()
{
        float a[5],sum1=0,sum2=1,sum3=0;
        int i,n=5;

        for(i=0;i<n;i++)
        {
                printf("\n Enter a[%d] : ",i);
                scanf("%f",&a[i]);

                sum1 = sum1 + a[i];
                sum2 = sum2 * a[i];                             // pow(sum2,1.0/n)
                sum3 = sum3 + (1.0/a[i]);
        }

        printf("\n Average = %f ",sum1/n);
        printf("\n Geometric Mean = %f ",pow(sum2,(1.0/n)));
        printf("\n Harmonic Mean = %f ",n/sum3);

        return 0;
}
```



```
 Enter a[0] : 10

 Enter a[1] : 20

 Enter a[2] : 30

 Enter a[3] : 40

 Enter a[4] : 50

 Average = 30.000000
 Geometric Mean = 26.051711
 Harmonic Mean = 21.897810
```

6) Write a C program to evaluate F(x) = x – x3 / 3! + x5 / 5 ! – x7 / 7! + … xn/ n!.

```
#include<stdio.h>
#include<math.h>
int Fact(int n);
void main()
{       int x,n,i,sign=1;
        float sum=0.0;

        printf("Enter x : ");
        scanf("%d",&x);
        printf("Enter n : ");
        scanf("%d",&n);

        printf("Sum of the series :  ");

        for (i=1 ; i<=n ; i=i+2)
        {
                if(sign==1)
                {
                        sum = sum + pow(x,i)/Fact(i);
                }
                else
                {
                        sum = sum - pow(x,i)/Fact(i);
                }
                //  ----------  Print Series  --------
                if (i == n  || i == n-1 )
                {
                        printf("%d^%d/%d! = %f \n",x,i,i,sum);
                }
                else
                {       if(sign==1)
                        {
                                printf("%d^%d/%d! - ",x,i,i);
                        }
                        else
                        {
                                printf("%d^%d/%d! + ",x,i,i);
                        }
                }
                sign = 1-sign;
        }
}
int Fact(int n)
{       int i,fact=1;
        for(i=1;i<=n;i++)
        {
                fact = fact * i;
        }
        return fact;
}
```



```
Enter x : 2
Enter n : 5
Sum of the series :  2^1/1! - 2^3/3! + 2^5/5! = 0.933333
```