

S.No: 1

Exp. Name: **Write a C program to display the Real number with 2 decimal places**

Date: 2022-12-30

Aim:

Write a C Program to display a real number with **2** decimal places.

Note: Do use printf() with '\n' at the end of printing output.

Sample Input and Output:

```
Enter a real number : 12.42536
The number after setting to 2 decimal places : 12.43
```

Page No. 1

ID: 22K61A05B4

2022-2026-CSE-B

Source Code:

```
realNumber.c

#include<stdio.h>
int main()
{
    float n;
    printf("Enter a real number : ");
    scanf("%f",&n);
    printf("The number after setting to 2 decimal places : %.2f\n",n);
}
```

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
Enter a real number :
12.42536
The number after setting to 2 decimal places : 12.43
```

Test Case - 2

User Output

```
Enter a real number :
9.12478
The number after setting to 2 decimal places : 9.12
```

Test Case - 3

User Output

```
Enter a real number :
3.96873
The number after setting to 2 decimal places : 3.97
```

Test Case - 4

User Output

Enter a real number :

8.90

The number after setting to 2 decimal places : 8.90

Test Case - 5

User Output

Enter a real number :

11.257

The number after setting to 2 decimal places : 11.26

Page No. 2

ID: 22K61A05B4

Sasi Institute of Technology and Engineering (Autonomous)

2022-2026-CSE-B

S.No: 2

Exp. Name: **Write a C program to convert the temperature from Fahrenheit to Celsius**

Date: 2022-12-30

Aim:

Write a program to convert the given temperature from **fahrenheit(F)** to **celsius(C)**.

During execution, the program should print the following message on the console:

Enter F :

For example, if the user gives the following as **input**:

Enter F : 32.45

Then the program should **print** the result as:

C = 0.250000

Note: Use the **printf()** function with a **newline** character (**\n**) at the end.

celsius = (5 / 9) * (fahrenheit - 32)

Source Code:

Program306.c

```
#include<stdio.h>
void main()
{
    float F,C;
    printf("Enter F : ");
    scanf("%f",&F);
    C = (5.0/9.0) * (F-32);
    printf("C = %f\n",C);
}
```

ID: 22K61A05B4

Page No. 3

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter F :

80.779

C = 27.099443

S.No: 3

Exp. Name: **Write a C program to convert the given temperature from Celsius to Fahrenheit**

Date: 2022-12-30

Aim:

Write a program to convert the given temperature from `celsius(C)` to `fahrenheit(F)`.

During execution, the program should print the following message on the console:

C:

For example, if the user gives the **input** as `1.23`:

C: 1.23

Then the program should **print** the result as:

F= 34.214001

Note: Use the `printf()` function with a **newline** character (`\n`) at the end.

The formula to find **fahrenheit** is `fahrenheit = celsius * (9.0 / 5.0) + 32.0`.

Source Code:

Program305.c

```
#include<stdio.h>
void main()
{
    float C,F;
    printf("enter C : " );
    scanf("%f",&C);
    F=((C*9.0)/5.0)+32.0;
    printf("F = %f\n",F);
}
```

Page No. 4

ID: 22K61A05B4

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

C:
0
F= 32.000000

Test Case - 2

User Output

C:
32
F= 89.599998

S.No: 4

Exp. Name: **Write a C program to find Area of a Triangle using Heron's formula**

Date: 2022-12-30

Aim:

Write a program to find the **area** of a **triangle** using Heron's formula.

During execution, the program should print the following message on the console:

sides:

For example, if the user gives the following as **input** (input is positive floating decimal point numbers):

sides: 2.3 2.4 2.5

Then the program should **print** the result round off upto 2 decimal places as:

area: 2.49

Instruction: Your input and output layout must match with the sample test cases (**values as well as text strings**).

The area of a triangle is given by $\text{Area} = \sqrt{p(p - a)(p - b)(p - c)}$, where p is half of the perimeter, or $(a + b + c) / 2$. Let a,b,c be the lengths of the sides of the given triangle.

Hint: Use **sqrt** function defined in **math.h** header file

Source Code:

Program313.c

```
#include<stdio.h>
#include<math.h>
void main()
{
    float a,b,c,s,A;
    printf("sides: ");
    scanf("%f%f%f",&a,&b,&c);
    s=(a+b+c)/2;
    A=sqrt(s*(s-a)*(s-b)*(s-c));
    printf("area: %.2f",A);
}
```

Page No. 5

ID: 22K61A05B4

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!**Test Case - 1****User Output**

sides:

2.3 2.4 2.5

area: 2.49

Test Case - 2

User Output

sides:

2.6 2.7 2.8

area: 3.15

Page No. 6

ID: 22K61A05B4

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 5

Exp. Name: **Write a C program to find the Largest and Smallest of Three numbers using Ternary operator**

Date: 2023-01-03

Aim:

Write a program to find the **largest** and **smallest** of the three given integers.

During execution, the program should print the following message on the console:

Enter 3 integers :

For example, if the user gives the following as **input**:

Enter 3 integers : 99 9 999

Then the program should **print** the result as:

999 is largest
9 is smallest

Note: Use the **printf()** function with a **newline** character (**\n**) at the end.

Source Code:

Program310.c

```
#include<stdio.h>
void main()
{
    int a,b,c;
    printf("Enter 3 integers: ");
    scanf("%d%d%d",&a,&b,&c);
    if(a>b&&a>c)
        printf("%d is largest\n",a);
    else if(b>c)
        printf("%d is largest\n",b);
    else
        printf("%d is largest\n",c);
    if(a<b&&a<c)
        printf("%d is smallest\n",a);
    else if(b<c)
        printf("%d is smallest\n",b);
    else
        printf("%d is smallest\n",c);
}
```

Page No. 7

ID: 22K61A05B4

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter 3 integers:

34 56 48

56 is largest
34 is smallest

Test Case - 2

User Output

Enter 3 integers:
56 34 12
56 is largest
12 is smallest

Test Case - 3

User Output

Enter 3 integers:
34 56 48
56 is largest
34 is smallest

S.No: 6

Exp. Name: **Write a C program to Swap Two numbers with out using a Third variable**

Date: 2022-12-30

Aim:

Write a program to **swap** two integer values without using a third variable.

During execution, the program should print the following message on the console:

Enter two integers :

For example, if the user gives the **input** as **99 89**:

Enter two integers : 99 89

Then the program should **print** the result as:

Before swapping a = 99, b = 89
After swapping a = 89, b = 99

Note: Use the **printf()** function with a **newline** character (**\n**) at the end.

Source Code:

Program311.c

```
#include<stdio.h>
main()
{
    int a,b;
    printf("Enter two integers: ");
    scanf("%d%d",&a,&b);
    printf("Before swapping a = %d, b = %d\n",a,b);
    a=a+b;
    b=a-b;
    a=a-b;
    printf("After swapping a = %d, b = %d\n",a,b);
}
```

Page No. 9

ID: 22K61A05B4

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter two integers:

23 67

Before swapping a = 23, b = 67

After swapping a = 67, b = 23

Test Case - 2

User Output

Enter two integers:

99 89

Before swapping a = 99, b = 89

After swapping a = 89, b = 99

Test Case - 3

User Output

Enter two integers:

77 78

Before swapping a = 77, b = 78

After swapping a = 78, b = 77

S.No: 7	Exp. Name: Write a C program to check whether the given number is Even or Odd using Bit-wise, Shift, and Arithmetic Operators	Date: 2023-01-03
---------	--	------------------

Aim:

Write a **C** Program to check whether the given number is **even** or **odd** using **bit-wise** operator, **shift** operator, and **arithmetic** operator.

Sample Input and Output:

```
Enter an integer : 233
Using bit-wise operator : 233 is an odd number
Using shift operator : 233 is an odd number
Using arithmetic operator : 233 is an odd number
```

ID: 22K61A05B4 Page No: 11

2022-2026-CSE-B

Source Code:

evenOrOdd.c

```
#include <stdio.h>
int main() {
    int number;
    printf("Enter an integer : ");
    scanf("%d",&number);
    printf("Using bit-wise operator : ");
    if (number&1) {
        printf("%d is an odd number\n", number);
    } else {
        printf("%d is an even number\n", number);
    }
    printf("Using shift operator : ");
    if ((number>>1)<<1==number) {
        printf("%d is an even number\n", number);
    } else {
        printf("%d is an odd number\n", number);
    }
    printf("Using arithmetic operator : ");
    if (number%2==0) {
        printf("%d is an even number\n", number);
    } else {
        printf("%d is an odd number\n", number);
    }
    return 0;
}
```

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter an integer :
12

Using bit-wise operator : 12 is an even number
Using shift operator : 12 is an even number
Using arithmetic operator : 12 is an even number

Test Case - 2

User Output

Enter an integer :
233
Using bit-wise operator : 233 is an odd number
Using shift operator : 233 is an odd number
Using arithmetic operator : 233 is an odd number

Test Case - 3

User Output

Enter an integer :
119
Using bit-wise operator : 119 is an odd number
Using shift operator : 119 is an odd number
Using arithmetic operator : 119 is an odd number

Test Case - 4

User Output

Enter an integer :
11111
Using bit-wise operator : 11111 is an odd number
Using shift operator : 11111 is an odd number
Using arithmetic operator : 11111 is an odd number

Test Case - 5

User Output

Enter an integer :
92345
Using bit-wise operator : 92345 is an odd number
Using shift operator : 92345 is an odd number
Using arithmetic operator : 92345 is an odd number

Test Case - 6

User Output

Enter an integer :
99
Using bit-wise operator : 99 is an odd number
Using shift operator : 99 is an odd number
Using arithmetic operator : 99 is an odd number

Test Case - 7

User Output

Enter an integer :

311

Using bit-wise operator : 311 is an odd number

Using shift operator : 311 is an odd number

Using arithmetic operator : 311 is an odd number

ID: 22K61A05B4

Page No: 13

Sasi Institute of Technology and Engineering (Autonomous)

2022-2026-CSE-B

S.No: 8	Exp. Name: Write a C program to find all Roots of a Quadratic equation	Date: 2023-01-03
---------	---	------------------

Aim:

Design and develop a flowchart or an algorithm that takes three coefficients (a, b, and c) of a **quadratic equation** ($ax^2+bx+c=0$) as input and computes all possible roots.

An equation is quadratic only if **a** is **non zero**.

If **a** is **zero** and **b** is **non zero** in the above equation then it becomes a **linear equation** ($bx + c = 0$).

If **a** and **b** are **zeros** then the it becomes a **constant equation**.

Implement a **C** program for the developed flowchart/algorithm and execute the same to output the possible roots for a given set of coefficients with appropriate messages.

At the time of execution, the program should print the message on the console as:

Enter coefficients a, b and c :

For example, if the user gives the **input** as:

Enter coefficients a, b and c : 2 6 4

then the program should **print** the result as:

The roots are real and distinct
root1 = -1.000000 and root2 = -2.000000

If the input is given as [0 0 0] then the result should be:

Invalid coefficients
Enter valid inputs

If the input is given as [0 2 8] then the result should be:

Linear equation
Root = -4.000000

If the input is given as [1 6 9] then the result should be:

The roots are real and equal
root1 = root2 = -3.000000

If the input is given as [1 4 7] then the result should be:

The roots are real and imaginary
root1 = -2.000000+i1.732051
root2 = -2.000000-i1.732051

Note - 1: Do use the **printf()** function with a **newline** character (`\n`) at the end.

Note - 2: Use **fabs()** funtion (**fabs(determinant)**) when the roots are real and imaginary.

Note - 3: Let us consider all the **coefficient** values as **float** values.

Source Code:

```
Program419.c

#include<stdio.h>
#include<math.h>
void main()
{
    float a,b,c,d,root1,root2,real,imag;
    printf("Enter coefficients a, b and c : ");
    scanf("%f%f%f",&a,&b,&c);
    d=b * b - 4 * a * c;
    if(a==0&&b==0&&c==0)
        printf("Invalid coefficients\nEnter valid inputs\n");
    else if(a==0)
    {
        root1=-c/b;
        printf("Linear equation\n");
        printf("Root = %f\n",root1);
    }
    else if(d>0)
    {
        root1=(-b+sqrt(d))/(2*a);
        root2=(-b-sqrt(d))/(2*a);
        printf("The roots are real and distinct\n");
        printf("root1 = %f and root2 = %f\n",root1,root2);
    }
    else if(d==0)
    {
        root1=root2=-b/(2*a);
        printf("The roots are real and equal\n");
        printf("root1 = root2 = %f\n",root1);
    }
    else
    {
        real=-b/(2*a);
        imag=sqrt(-d)/(2*a);
        printf("The roots are real and imaginary\n");
        printf("root1 = %f+i%f\nroot2 = %f-i%f\n",real,imag,real,imag);
    }
}
```

ID: 22K61A05B4 Page No: 15

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter coefficients a, b and c :
2 6 4
The roots are real and distinct
root1 = -1.000000 and root2 = -2.000000

Test Case - 2

User Output

Enter coefficients a, b and c :

0 0 0

Invalid coefficients

Enter valid inputs

Test Case - 3

User Output

Enter coefficients a, b and c :

0 2 8

Linear equation

Root = -4.000000

Test Case - 4

User Output

Enter coefficients a, b and c :

1 6 9

The roots are real and equal

root1 = root2 = -3.000000

Test Case - 5

User Output

Enter coefficients a, b and c :

1 -5 3

The roots are real and distinct

root1 = 4.302776 and root2 = 0.697224

Test Case - 6

User Output

Enter coefficients a, b and c :

1 4 7

The roots are real and imaginary

root1 = -2.000000+i1.732051

root2 = -2.000000-i1.732051

S.No: 9	Exp. Name: Write a C program to display Grade based on 6 subject marks using if-else-if	Date: 2023-01-03
---------	--	------------------

Aim:

Write a C Program to display [grade] based on [6] subject marks using an if-else-if ladder.

marks \geq 90% is grade A

marks \geq 80% and $<$ 90% is grade B.

marks \geq 70% and $<$ 80% is grade C.

marks \geq 60% and $<$ 70% is grade D.

marks \geq 40% and $<$ 60% is grade E.

marks $<$ 40% is grade Fail.

Sample Input and Output:

```
Enter the six subjects marks : 60 50 70 90 55 69
Total marks : 394
Percentage : 65.666664
Grade : D
```

Source Code:

```
grade.c
```

ID: 22K61A05B4 Page No: 17

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include<stdio.h>
int main()
{
int s1,s2,s3,s4,s5,s6,total;
float avg;
printf("Enter the six subjects marks : ");
scanf("%d%d%d%d%d",&s1,&s2,&s3,&s4,&s5,&s6);
total=s1+s2+s3+s4+s5+s6;
avg=(float) total/6;
printf("Total marks : %d\n",total);
printf("Percentage : %f\n",avg);
if(avg<40)
printf("Grade : Fail\n");
else if(avg<60)
printf("Grade : E\n");
else if(avg<70)
printf("Grade : D\n");
else if(avg<80)
printf("Grade : C\n");
else if(avg<90)
printf("Grade : B\n");
else
printf("Grade : A\n");

}

```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter the six subjects marks :
60 50 70 90 55 69
Total marks : 394
Percentage : 65.666664
Grade : D

Test Case - 2

User Output

Enter the six subjects marks :
100 90 28 45 33 80
Total marks : 376
Percentage : 62.666668
Grade : D

Test Case - 3

User Output

Enter the six subjects marks :

90 89 85 97 79 88

Total marks : 528

Percentage : 88.000000

Grade : B

ID: 22K61A05B4 Page No: 19

Test Case - 4**User Output**

Enter the six subjects marks :

20 28 30 25 33 38

Total marks : 174

Percentage : 29.000000

Grade : Fail

2022-2026-CSE-B

Test Case - 5**User Output**

Enter the six subjects marks :

65 70 75 60 80 85

Total marks : 435

Percentage : 72.500000

Grade : C

S.No: 10	Exp. Name: <i>Write a C program to make a simple Calculator to Add, Subtract, Multiply or Divide using switch-case</i>	Date: 2023-01-30
----------	---	------------------

Aim:

Write a program to read two integer values and an arithmetic operator, depending on the operator perform different arithmetic operations.

If integer values **2** and **3** are given with operator **+**, then the output should be $2 + 3 = 5$.

If integer values **6** and **3** are given with operator **/**, then the output should be $6 / 3 = 2$.

If other than arithmetic operator is given, then display "**Error! Operator is not correct**".

Note: Space before %c removes any white space (blanks, tabs, or newlines). It means %c without space will read white space like new line(\n), spaces(' ') or tabs(\t). By adding space before %c, we are skipping this and reading only the char given.

Instruction: To run your custom test cases strictly map your input and output layout with the visible test cases.

Source Code:

Program406.c

```
#include<stdio.h>
int main()
{
    int n,m;
    char c;
    printf("Values: ");
    scanf("%d%d",&n,&m);
    printf("Operator: ");
    getchar();
    scanf("\n%c",&c);
    switch(c)
    {
        case '+': printf("%d + %d = %d\n",n,m,n+m);
        break;
        case '-': printf("%d - %d = %d\n",n,m,n-m);
        break;
        case '*': printf("%d * %d = %d\n",n,m,n*m);
        break;
        case '/': if(m==0)
                    printf("Division is not possible! Divide by zero error\n");
                else
                    printf("%d / %d = %d\n",n,m,n/m);
                break;
        case '%': if(m==0)
                    printf("Modulo division is not possible! Divide by zero error\n");
                else
                    printf("%d %% %d = %d\n",n,m,n%m);
                break;
        default: printf("Invalid Operator\n");
    }
}
```

ID: 22K61A05B4 Page No: 20

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1	
User Output	
Values:	
6 9	
Operator:	
-	
6 - 9 = -3	

Test Case - 2	
User Output	
Values:	
6 9	
Operator:	
*	
6 * 9 = 54	

Test Case - 3	
User Output	
Values:	
8 9	
Operator:	
@	
Invalid Operator	

Test Case - 4	
User Output	
Values:	
12 0	
Operator:	
/	
Division is not possible! Divide by zero error	

Test Case - 5	
User Output	
Values:	
5 0	
Operator:	
%	
Modulo division is not possible! Divide by zero error	

ID: 22K61A05B4 Page No: 21

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 11

Exp. Name: **Write a C program to find the number of 0's and 1's in a Binary representation of a given number**

Date: 2023-01-18

Aim:

Write a **C** Program to count the number of **0's** and **1's** in a **binary** representation of a given number.

Sample Input and Output:

```
Enter a decimal number : 25
Binary number : 11001
Number of zero's : 2
Number of one's : 3
```

Source Code:

zerosOnesCount.c

```
#include<stdio.h>
#include<math.h>
int main()
{
    int num,b_num=0,once_count=0,zero_count=0,count=0;
    printf("Enter a decimal number : ");
    scanf("%d",&num);
    while(num!=0)
    {
        int rem=num%2;
        if(rem==0)
            zero_count++;
        else
            once_count++;
        int c=pow(10,count);
        b_num=b_num+rem*c;
        num=num/2;
        count++;
    }
    printf("Binary number : %d\n",b_num);
    printf("Number of zero's : %d\n",zero_count);
    printf("Number of one's : %d\n",once_count);
}
```

ID: 22K61A05B4 Page No: 22

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
Enter a decimal number :
10
Binary number : 1010
Number of zero's : 2
Number of one's : 2
```

Test Case - 2

User Output

Enter a decimal number :

7

Binary number : 111

Number of zero's : 0

Number of one's : 3

Page No:23

ID: 22K61A05B4

Test Case - 3

User Output

Enter a decimal number :

4

Binary number : 100

Number of zero's : 2

Number of one's : 1

2022-2026-CSE-B

Test Case - 4

User Output

Enter a decimal number :

25

Binary number : 11001

Number of zero's : 2

Number of one's : 3

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 5

User Output

Enter a decimal number :

255

Binary number : 11111111

Number of zero's : 0

Number of one's : 8

Test Case - 6

User Output

Enter a decimal number :

201

Binary number : 11001001

Number of zero's : 4

Number of one's : 4

Test Case - 7

User Output

Enter a decimal number :

111

Binary number : 1101111

Number of zero's : 1

Number of one's : 6

Test Case - 8**User Output**

Enter a decimal number :

99

Binary number : 1100011

Number of zero's : 3

Number of one's : 4

Page No: 24

ID: 22K61A05B4

Sasi Institute of Technology and Engineering (Autonomous)

2022-2026-CSE-B

S.No: 12

Exp. Name: **Write a C program to find all the Prime numbers between the given limits**

Date: 2023-01-18

Aim:

Write a sample code to find all the **prime numbers** between the limits.

[Hint: A **prime number** is a positive integer greater than 1 and which is divisible by 1 and itself only. A few prime numbers are 2, 3, 5, 7, 11, 13, 17, 19, etc.]

At the time of execution, the program should print the message on the console as:

Enter lower and upper limits :

For example, if the user gives the **input** as:

Enter lower and upper limits : 10 20

then the program should **print** the result as:

Prime numbers between 10 and 20 are : 11 13 17 19

Note: Do use the **printf()** function with **spaces** before and after the conversional string.

Source Code:

Program12.c

```
#include<stdio.h>
int main()
{
    int l,u,i,j,count=0;
    printf("Enter lower and upper limits : ");
    scanf("%d%d",&l,&u);
    printf("Prime numbers between %d and %d are : ",l,u);
    for(i=l;i<=u;i++)
    {
        count=0;
        for(j=1;j<=i;j++)
        {
            if(i%j==0)
                count++;
        }
        if(count==2)
        {
            printf("%d ",i);
        }
    }
}
```

ID: 22K61A05B4 Page No: 25

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter lower and upper limits :

3 20

Prime numbers between 3 and 20 are : 3 5 7 11 13 17 19

Test Case - 2**User Output**

Enter lower and upper limits :

11 29

Prime numbers between 11 and 29 are : 11 13 17 19 23 29

ID: 22K61A05B4 Page No: 26

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 13

Exp. Name: **Write a Program to print Multiplication Table for a given number**

Date: 2022-12-30

Aim:

Write a program to print the [multiplication table](#) for a given number with the number of rows in the table.

For example, for a number [2](#) with [3](#) rows, the output should be:

```
2 * 1 = 2  
2 * 2 = 4  
2 * 3 = 6
```

At the time of execution, the program should print the following messages one by one on the console as:

```
Enter an integer number :  
Enter number of rows :
```

For example, if the user gives the **input** as:

```
Enter an integer number : 5  
Enter number of rows : 4
```

then the program should **print** the result as:

```
5 * 1 = 5  
5 * 2 = 10  
5 * 3 = 15  
5 * 4 = 20
```

Note: Do use the **printf()** function with a **newline** character ([\n](#)).

Source Code:

Program411.c

```
#include<stdio.h>  
int main()  
{  
    int n,i,r;  
    printf("Enter an integer number : ");  
    scanf("%d",&n);  
    printf("Enter number of rows : ");  
    scanf("%d",&r);  
    for(i=1;i<=r;i++)  
    {  
        printf("%d * %d = %d\n",n,i,n*i);  
    }  
}
```

ID: 22K61A05B4 Page No: 27

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter an integer number :

3

Enter number of rows :

6

3 * 1 = 3

3 * 2 = 6

3 * 3 = 9

3 * 4 = 12

3 * 5 = 15

3 * 6 = 18

Test Case - 2

User Output

Enter an integer number :

5

Enter number of rows :

4

5 * 1 = 5

5 * 2 = 10

5 * 3 = 15

5 * 4 = 20

Test Case - 3

User Output

Enter an integer number :

12

Enter number of rows :

7

12 * 1 = 12

12 * 2 = 24

12 * 3 = 36

12 * 4 = 48

12 * 5 = 60

12 * 6 = 72

12 * 7 = 84

Test Case - 4

User Output

Enter an integer number :

15

Enter number of rows :

10

15 * 1 = 15

15 * 2 = 30

15 * 3 = 45

15 * 4 = 60

15 * 5 = 75
15 * 6 = 90
15 * 7 = 105
15 * 8 = 120
15 * 9 = 135
15 * 10 = 150

S.No: 14

Exp. Name: **Write a C program to find whether the given number is Armstrong or not**

Date: 2023-01-20

Aim:

Write a sample code to check whether the given number is an [armstrong number](#) or not.

Hint: An [armstrong number](#) is a number that is the sum of its own digits each raised to the power of the number of digits.

For example,

$$9 = 9^1 = 9$$

$$371 = 3^3 + 7^3 + 1^3 = 27 + 343 + 1 = 371$$

$$8208 = 8^4 + 2^4 + 0^4 + 8^4 = 4096 + 16 + 0 + 4096 = 8208$$

At the time of execution, the program should print the message on the console as:

Enter any number :

For example, if the user gives the **input** as:

Enter any number : 153

then the program should **print** the result as:

The given number 153 is an armstrong number

Similarly, if the input is given as 121 then the output should be "**The given number 121 is not an armstrong number**".

Note: Do use the **printf()** function with a **newline** character ([\n](#)) at the end.

Source Code:

Program410.c

ID: 22K61A05B4 Page No: 30

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include<stdio.h>
#include<math.h>
int main ()
{
    int num,count=0,arm=0,num2,rem;
    printf("Enter any number : ");
    scanf("%i",&num);
    int num1=num;
    while(num1!=0)
    {
        num1=num1/10;
        count++;
    }
    num2=num;
    while(num2!=0)
    {
        rem=num2%10;
        arm=arm+pow(rem,count);
        num2=num2/10;
    }
    if(num==arm)
    {
        printf("The given number %d is an armstrong number\n",num);
    }
    else
    {
        printf("The given number %d is not an armstrong number\n",num);
    }
    return 0;
}

```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter any number :

370

The given number 370 is an armstrong number

Test Case - 2

User Output

Enter any number :

1824

The given number 1824 is not an armstrong number

Test Case - 3

User Output

Enter any number :

5

The given number 5 is an armstrong number

Test Case - 4

User Output

Enter any number :

1634

The given number 1634 is an armstrong number

ID: 22K61A05B4 Page No: 32

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 15

Exp. Name: **Write a C program to calculate whether a given number is Palindrome or not**

Date: 2023-01-18

Aim:

Write a program to find the [reverse](#) of an integer number and check whether it is [Palindrome](#) or not.

At the time of execution, the program should print the message on the console as:

Enter an integer :

For example, if the user gives the [input](#) as:

Enter an integer : 2014

then the program should [print](#) the result as:

The reverse of a given number : 4102
2014 is not a palindrome

If the input is given as [1221](#) then the result should be:

The reverse of a given number : 1221
1221 is a palindrome

Source Code:

Program421.c

```
#include<stdio.h>
int main()
{
    int n,rev=0,temp,rem;
    printf("Enter an integer : ");
    scanf("%d",&n);
    temp=n;
    while(n>0)
    {
        rem=n%10;
        rev=rev+(rev*10);
        n=n/10;
    }
    if (rev==temp)
    {
        printf("The reverse of a given number : %d\n",rev);
        printf("%d is a palindrome\n",temp);
    }
    else
    {
        printf("The reverse of a given number : %d\n",rev);
        printf("%d is not a palindrome\n",temp);
    }
    return 0;
}
```

ID: 22K61A05B4 Page No: 33

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter an integer :
2017
The reverse of a given number : 7102
2017 is not a palindrome

ID: 22K61A05B4 Page No: 34

Test Case - 2
User Output
Enter an integer :
1221
The reverse of a given number : 1221
1221 is a palindrome

2022-2026-CSE-B

Test Case - 3
User Output
Enter an integer :
12321
The reverse of a given number : 12321
12321 is a palindrome

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 4
User Output
Enter an integer :
18771
The reverse of a given number : 17781
18771 is not a palindrome

S.No: 16

Exp. Name: **Write a C program to find the Sum of individual digits of a given number**

Date: 2023-01-18

Aim:

Write a program to find the **sum of individual digits** of a given number.

At the time of execution, the program should print the message on the console as:

Enter an integer :

For example, if the user gives the **input** as **1234**:

Enter an integer : 1234

then the program should **print** the result as:

The sum of digits of the given number 1234 = 10

Note: Do use the **printf()** function with a **newline** character (**\n**) at the end.

Source Code:

LoopExample4.c

```
#include<stdio.h>
int main()
{
    int i,sum=0,rem;
    printf("Enter an integer : ");
    scanf("%d",&i);
    int num=i;
    while(num!=0)
    {
        rem=num%10;
        sum=sum+rem;
        num=num/10;
    }
    printf("The sum of digits of the given number %d = %d\n",i,sum);
}
```

ID: 22K61A05B4 Page No: 35

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter an integer :

345

The sum of digits of the given number 345 = 12

Test Case - 2

User Output

Enter an integer :

6543

The sum of digits of the given number 6543 = 18

Test Case - 3

User Output

Enter an integer :

96584

The sum of digits of the given number 96584 = 32

S.No: 17

Exp. Name: **Write a C program to calculate the series 1 +
2 + 3 + 4 + + n**

Date: 2023-01-19

Aim:

Write a C program to calculate the series **1 + 2 + 3 + 4 + + n**.

Sample Input and Output:

```
Enter n value : 10  
Sum of 10 natural numbers : 55
```

Source Code:

series1.c

```
#include<stdio.h>  
void main()  
{  
    int n,i=1,sum=0;  
    printf("Enter n value : ");  
    scanf("%d",&n);  
    while(i<=n)  
    {  
        sum=sum+i;  
        i++;  
    }  
    printf("Sum of %d natural numbers : %d\n",n,sum);  
}
```

ID: 22K61A05B4 Page No: 37

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter n value :

10

Sum of 10 natural numbers : 55

Test Case - 2

User Output

Enter n value :

14

Sum of 14 natural numbers : 105

Test Case - 3

User Output

Enter n value :

11

Sum of 11 natural numbers : 66

Test Case - 4

User Output

Enter n value :

8

Sum of 8 natural numbers : 36

ID: 22K61A05B4 Page No: 38

Test Case - 5

User Output

Enter n value :

99

Sum of 99 natural numbers : 4950

2022-2026-CSE-B

Test Case - 6

User Output

Enter n value :

67

Sum of 67 natural numbers : 2278

S.No: 18

Exp. Name: **Write a C program to evaluate $1 + 1/2 + 1/3 + \dots + 1/n$**

Date: 2023-01-03

Aim:

Write a C program to evaluate $1 + 1/2 + 1/3 + \dots + 1/n$.

At the time of execution, the program should print the message on the console as:

Enter n value :

For example, if the user gives the input as:

Enter n value : 2

then the program should print the result as:

Result : 1.500000

ID: 22K61A05B4 Page No: 39

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Source Code:

SumOfSeries10.c

```
#include<stdio.h>
void main()
{
    int limit;
    float sum=0,count;
    printf("Enter n value : ");
    scanf("%d",&limit );
    for(count=1;count<=limit;count++)
    {
        sum=sum+ (1/count);
    }
    printf("Result : %f\n",sum);
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter n value :

2

Result : 1.500000

Test Case - 2

User Output

Enter n value :

10

Result : 2.928968

Test Case - 3

User Output

Enter n value :

25

Result : 3.815958

Test Case - 4

User Output

Enter n value :

30

Result : 3.994987

Test Case - 5

User Output

Enter n value :

99

Result : 5.177378

Test Case - 6

User Output

Enter n value :

999

Result : 7.484478

Test Case - 7

User Output

Enter n value :

1

Result : 1.000000

Test Case - 8

User Output

Enter n value :

5

Result : 2.283334

S.No: 19	Exp. Name: Write a C program to compute the sum of this geometric progression: $1+x+x^2+x^3+\dots+x^n$.	Date: 2023-01-18
----------	--	------------------

Aim:

Write a **C** program to read in two numbers, x and n , and then compute the sum of this geometric progression:
 $1+x+x^2+x^3+\dots+x^n$.

For example: if n is 3 and x is 5, then the program computes **1+5+25+125**.

At the time of execution, the program should print the message on the console as:

Enter x value :

For example,

if the user gives the **input** as:

Enter x value : 3

Now, the program should print the message on the console as:

Enter n value :

For example, if the user gives the **input** as:

Enter n value : 5

then the program should **print** the result as:

Sum of the series $1 + x + \dots + x^5 = 364$

ID: 22K61A05B4 Page No: 41

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Source Code:

SumOfSeries.c

```
#include<stdio.h>
#include<math.h>
void main()
{
    int x,n,i=1,sum=1,result;
    printf("Enter x value : ");
    scanf("%d",&x);
    printf("Enter n value : ");
    scanf("%d",&n);
    while(i<=n)
    {
        result=pow(x,i);
        sum=sum+result;
        i++;
    }
    printf("Sum of the series 1 + x + \dots + x ^ %d = %d\n",n,sum);
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter x value :

3

Enter n value :

5

Sum of the series $1 + x + \dots + x^5 = 364$ **Test Case - 2****User Output**

Enter x value :

2

Enter n value :

5

Sum of the series $1 + x + \dots + x^5 = 63$

S.No: 20

Exp. Name: **Write a C program to interchange the Largest and Smallest numbers in the array**

Date: 2023-01-18

Aim:

Write a C program to **interchange** the **largest** and **smallest** numbers in the array.

Sample Input and Output:

```
Enter the size of array : 5
Enter 5 integers: 11 44 66 22 99
Interchanging largest value 99 and smallest value 11
After interchange, array elements are: 99 44 66 22 11
```

Source Code:

interchange.c

```
#include<stdio.h>
void main()
{
    int a[100],size,max,min,maxpos,minpos,i,temp;
    printf("Enter the size of array : ");
    scanf("%d",&size);
    printf("Enter %d integers: ",size);
    for(i=0;i<size;i++)
        scanf("%d",&a[i]);
    max=a[0];
    min=a[0];
    maxpos=0;
    minpos=0;
    for(i=1;i<size;i++)
    {
        if(a[i]>max)
        {
            max=a[i];
            maxpos=i;
        }
        if(a[i]<min)
        {
            min=a[i];
            minpos=i;
        }
    }
    printf("Interchanging largest value %d and smallest value %d",a[maxpos],a[minpos]);
    temp=a[maxpos];
    a[maxpos]=a[minpos];
    a[minpos]=temp;
    printf("\nAfter interchange, array elements are: ");
    for(i=0;i<size;i++)
        printf("%d ",a[i]);

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

ID: 22K61A05B4 Page No: 43

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
Enter the size of array :  
5  
Enter 5 integers:  
11 44 66 22 99  
Interchanging largest value 99 and smallest value 11  
After interchange, array elements are: 99 44 66 22 11
```

Test Case - 2

User Output

```
Enter the size of array :  
3  
Enter 3 integers:  
215 315 4  
Interchanging largest value 315 and smallest value 4  
After interchange, array elements are: 215 4 315
```

Test Case - 3

User Output

```
Enter the size of array :  
2  
Enter 2 integers:  
99 999  
Interchanging largest value 999 and smallest value 99  
After interchange, array elements are: 999 99
```

Test Case - 4

User Output

```
Enter the size of array :  
4  
Enter 4 integers:  
101 99 201 401  
Interchanging largest value 401 and smallest value 99  
After interchange, array elements are: 101 401 201 99
```

S.No: 21

Exp. Name: **Write a C program to Search an element using Linear Search process**

Date: 2023-01-18

Aim:

Write a program to [search](#) a key element with in the given array of elements using [linear search](#) process.

At the time of execution, the program should print the message on the console as:

Enter value of n :

For example, if the user gives the **input** as:

Enter value of n : 3

Next, the program should print the messages one by one on the console as:

Enter element for a[0] :
Enter element for a[1] :
Enter element for a[2] :

if the user gives the **input** as:

Enter element for a[0] : 89
Enter element for a[1] : 33
Enter element for a[2] : 56

Next, the program should print the message on the console as:

Enter key element :

if the user gives the **input** as:

Enter key element : 56

then the program should **print** the result as:

The key element 56 is found at the position 2

Similarly if the key element is given as **25** for the above one dimensional array elements then the program should print the output as "**The Key element 25 is not found in the array**".

Note: Do use the **printf()** function with a **newline** character ([\n](#)) at the end.

Source Code:

Program509.c

ID: 22K61A05B4 Page No: 45

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include<stdio.h>
void main()
{
    int a[20],i,x,n;
    printf("Enter value of n : ");
    scanf("%d",&n);
    for(i=0;i<n;++i)
    {
        printf("Enter element for a[%d] : ",i);
        scanf("%d",&a[i]);
    }
    printf("Enter key element : ");
    scanf("%d",&x);
    for(i=0;i<n;++i)
    if(a[i]==x)
    break;
    if(i<n)
    printf("The key element %d is found at the position %d",x,i);
    else
    printf("The key element %d is not found in the array",x);
    printf("\n");
}

```

ID: 22K61A05B4 Page No: 46

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter value of n :
5
Enter element for a[0] :
45
Enter element for a[1] :
67
Enter element for a[2] :
35
Enter element for a[3] :
28
Enter element for a[4] :
16
Enter key element :
28
The key element 28 is found at the position 3

Test Case - 2
User Output
Enter value of n :
5
Enter element for a[0] :

```
2
Enter element for a[1] :
7
Enter element for a[2] :
5
Enter element for a[3] :
1
Enter element for a[4] :
4
Enter key element :
2
The key element 2 is found at the position 0
```

ID: 22K61A05B4 Page No: 47

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 3

User Output

```
Enter value of n :
4
Enter element for a[0] :
452
Enter element for a[1] :
356
Enter element for a[2] :
754
Enter element for a[3] :
127
Enter key element :
127
The key element 127 is found at the position 3
```

Test Case - 4

User Output

```
Enter value of n :
3
Enter element for a[0] :
5
Enter element for a[1] :
7
Enter element for a[2] :
3
Enter key element :
4
The key element 4 is not found in the array
```

Test Case - 5

User Output

```
Enter value of n :  
3  
Enter element for a[0] :  
11  
Enter element for a[1] :  
45  
Enter element for a[2] :  
37  
Enter key element :  
25  
The key element 25 is not found in the array
```

ID: 22K61A05B4 Page No: 48

Sasi Institute of Technology and Engineering (Autonomous)

2022-2026-CSE-B

S.No: 22

Exp. Name: **Write a C program to print the Pattern using a Character Array**

Date: 2023-01-06

Aim:

Write a **C** program to print the **pattern** using a character array.

Sample Input and Output:

```
Enter the string : SASI
S
SA
SAS
SASI
```

Source Code:

stringPattern.c

```
#include<stdio.h>
#include<string.h>
void main()
{
    int i,j,k;
    char sasi[100];
    printf("Enter the string : ");
    scanf("%s",sasi);
    k=strlen(sasi);
    for(i=0;i<k;i++)
    {
        for(j=0;j<=i;j++)
        {
            printf("%c",sasi[j]);
        }
        printf("\n");
    }
}
```

ID: 22K61A05B4 Page No: 49

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
Enter the string :
SASI
S
SA
SAS
SASI
```

Test Case - 2

User Output

Enter the string :

CodeTantra

C

Co

Cod

Code

CodeT

CodeTa

CodeTan

CodeTant

CodeTantr

CodeTantra

Test Case - 3**User Output**

Enter the string :

SARASWATHI

S

SA

SAR

SARA

SARAS

SARASW

SARASWA

SARASWAT

SARASWATH

SARASWATHI

Test Case - 4**User Output**

Enter the string :

RoboticTool

R

Ro

Rob

Robo

Robot

Roboti

Robotic

RoboticT

RoboticTo

RoboticToo

RoboticTool

S.No: 23

Exp. Name: **Write a C program to find the Addition of Two matrices by checking compatibility**

Date: 2023-01-30

Aim:

Write a program to find the addition of two matrices.

Note :

- Do use the **printf()** function with a **newline** character (\n).
- If the sizes of two matrices are not equal then print **Addition is not possible** and stop the process.

Source Code:

Program511.c

ID: 22K61A05B4 Page No: 51

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include<stdio.h>
int main()
{
    int a[100][100],b[100][100],i,j,r1,c1,r2,c2,sum[100][100];
    printf("Size of mat1: ");
    scanf("%d%d", &r1,&c1);
    printf("mat1: ",r1*c1);
    for(i=0;i<r1;++i)
        for(j=0;j<c1;++j)
            scanf("%d",&a[i][j]);
    printf("Size of mat2: ");
    scanf("%d%d",&r2,&c2);
    printf("mat2: ",r2*c2);
    for(i=0;i<r2;++i)
        for(j=0;j<c2;++j)
            scanf("%d",&b[i][j]);
    if(r1==r2&&c1==c2)
    {
        for (i = 0;i < r1;++i)
            for (j = 0;j < c1;++j)
                sum[i][j] = a[i][j] + b[i][j];
        printf("mat1\n");
        for (i=0;i<r1;++)
        {
            for (j=0;j<c1;++)
            {
                printf("%d ",a[i][j]);
            }
            printf("\n");
        }
        printf("mat2\n");
        for (i=0;i<r2;++)
        {
            for (j=0;j<c2;j++)
            {
                printf("%d ",b[i][j]);
            }
            printf("\n");
        }
        printf("Addition\n");
        for (i = 0; i < r1;i++)
        {
            for(j = 0; j < c1;j++)
            {
                printf("%d ", sum[i][j]);
            }
            printf("\n");
        }
    }
    else
    {
        printf("mat1\n");
        for (i=0;i<r1;++)
    }
}

```

```

    {
        printf("%d ",a[i][j]);
    }
    printf("\n");
}
printf("mat2\n");
for (i=0;i<r2;++i)
{
    for (j=0;j<c2;++j)
    {
        printf("%d ",b[i][j]);
    }
    printf("\n");
}
printf("Addition is not possible\n");
}
return 0;
}

```

ID: 22K61A05B4 Page No: 53

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Size of mat1:
2 2
mat1:
11 22 33 44
Size of mat2:
2 2
mat2:
22 33 44 55
mat1
11 22
33 44
mat2
22 33
44 55
Addition
33 55
77 99

Test Case - 2
User Output
Size of mat1:
2 3
mat1:
1 2 3 4 5 6

```
Size of mat2:  
3 2  
mat2:  
1 3 4 5 6 7  
mat1  
1 2 3  
4 5 6  
mat2  
1 3  
4 5  
6 7  
Addition is not possible
```

S.No: 24

Exp. Name: **Write a C program to find the Multiplication of Two matrices by checking compatibility**

Date: 2023-01-29

Aim:

Write a program to find the [multiplication of two matrices](#).

At the time of execution, the program should print the message on the console as:

Enter the row & column sizes of matrix-1 :

For example, if the user gives the **input** as:

Enter the row & column sizes of matrix-1 : 3 2

Next, the program should print the message on the console as:

Enter matrix-1 6 elements :

if the user gives the **input** as:

Enter matrix-1 6 elements : 1 2 3 4 5 6

Next, the program should print the message on the console as:

Enter the row & column sizes of matrix-2 :

if the user gives the **input** as:

Enter the row & column sizes of matrix-2 : 2 3

Next, the program should print the message on the console as:

Enter matrix-2 6 elements :

if the user gives the **input** as:

Enter matrix-2 6 elements : 4 5 6 7 8 9

then the program should **print** the result as:

```
The given matrix-1 is  
1 2 3  
4 5 6  
The given matrix-2 is  
4 5  
6 7  
8 9  
Multiplication of two matrices is  
40 46  
94 109
```

Note: 1 Do use the **printf()** function with a **newline** character (`\n`).

Note: 2 Display **Multiplication is not possible** if multiplication operation can not be performed.

Source Code:

Program513.c

```

#include<stdio.h>
int main()
{
    int r1,c1,r2,c2,a[100][100],b[100][100],mul[100][100],i,j,k;
    printf("Enter the row & column sizes of matrix-1 : ");
    scanf("%d%d",&r1,&c1);
    printf("Enter matrix-1 %d elements : ",r1*c1);
    for(i=0;i<r1;i++)
        for(j=0;j<c1;j++)
            scanf("%d",&a[i][j]);
    printf("Enter the row & column sizes of matrix-2 : ");
    scanf("%d%d",&r2,&c2);
    printf("Enter matrix-2 %d elements : ",r2*c2);
    for(i=0;i<r2;i++)
        for(j=0;j<c2;j++)
            scanf("%d",&b[i][j]);
    printf("The given matrix-1 is\n");
    for(i=0;i<r1;i++)
    {
        for(j=0;j<c1;j++)
        {
            printf("%d ",a[i][j]);
        }
        printf("\n");
    }
    printf("The given matrix-2 is\n");
    {
        for(i=0;i<r2;i++)
        {
            for(j=0;j<c2;j++)
            {
                printf("%d ",b[i][j]);
            }
            printf("\n");
        }
        if(c1==r2)
        {
            for(i=0;i<r1;i++)
                for(j=0;j<c2;j++)
                    mul[i][j]=0;
            for(i=0;i<r1;i++)
                for(j=0;j<c2;j++)
                    for(k=0;k<r2;k++)
                        mul[i][j]+=a[i][k]*b[k][j];
            printf("Multiplication of two matrices is\n");
            for(i=0;i<r1;i++)
            {
                for(j=0;j<c2;j++)
                {
                    printf("%d ",mul[i][j]);
                }
                printf("\n");
            }
        }
    }
}
else

```

```
        }
    }
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter the row & column sizes of matrix-1 :
2 2
Enter matrix-1 4 elements :
11 33 22 44
Enter the row & column sizes of matrix-2 :
2 2
Enter matrix-2 4 elements :
11 33 44 22
The given matrix-1 is
11 33
22 44
The given matrix-2 is
11 33
44 22
Multiplication of two matrices is
1573 1089
2178 1694

Test Case - 2
User Output
Enter the row & column sizes of matrix-1 :
2 3
Enter matrix-1 6 elements :
1 2 3 4 5 6
Enter the row & column sizes of matrix-2 :
3 2
Enter matrix-2 6 elements :
1 2 3 4 5 6
The given matrix-1 is
1 2 3
4 5 6
The given matrix-2 is
1 2
3 4
5 6
Multiplication of two matrices is
22 28
49 64

Test Case - 3

User Output

Enter the row & column sizes of matrix-1 :

2 3

Enter matrix-1 6 elements :

1 2 3 4 5 6

Enter the row & column sizes of matrix-2 :

2 2

Enter matrix-2 4 elements :

1 2 3 4

The given matrix-1 is

1 2 3

4 5 6

The given matrix-2 is

1 2

3 4

Multiplication is not possible

ID: 22K61A05B4 Page No: 58

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 4

User Output

Enter the row & column sizes of matrix-1 :

3 3

Enter matrix-1 9 elements :

11 22 33 44 55 66 77 88 99

Enter the row & column sizes of matrix-2 :

3 3

Enter matrix-2 9 elements :

99 88 77 66 55 44 33 22 11

The given matrix-1 is

11 22 33

44 55 66

77 88 99

The given matrix-2 is

99 88 77

66 55 44

33 22 11

Multiplication of two matrices is

3630 2904 2178

10164 8349 6534

16698 13794 10890

S.No: 25

Exp. Name: **Write a Program to check whether the given Matrix is Symmetric or not**

Date: 2023-01-20

Aim:

Write a **C** program to find whether a given matrix is a [symmetric matrix](#) or not.

Hint: A **symmetric matrix** is a square matrix that is equal to its **transpose**.

At the time of execution, the program should print the message on the console as:

Enter the order of matrix :

For example, if the user gives the **input** as:

Enter the order of matrix : 2 2

Next, the program should print the message on the console as:

Enter 4 elements :

if the user gives the **input** as:

Enter 4 elements : 4 5 5 4

then the program should **print** on the console as:

The given matrix is
4 5
5 4
Transpose of the given matrix is
4 5
5 4
The given matrix is symmetric matrix

If the condition is **true**, then the program should **print** the result as :

The given matrix is symmetric matrix

Otherwise, the program should **print** the result as :

The given matrix is not symmetric matrix

Note: Do use the **printf()** function with a **newline** character (**\n**).

Source Code:

SymmetricMatrix.c

ID: 22K61A05B4 Page No: 59

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include<stdio.h>
int main()
{
    int m, n, c, d, matrix[10][10], transpose[10][10];
    printf("Enter the order of matrix : ");
    scanf("%d%d", &m, &n);
    printf("Enter %d elements : ", m*n);
    for (c = 0; c < m; c++)
        for (d = 0; d < n; d++)
            scanf("%d", &matrix[c][d]);
    printf("The given matrix is\n");
    for (c = 0; c < m; c++)
    {
        for (d = 0; d < n; d++)
        {
            printf("%d ", matrix[c][d]);
        }
        printf("\n");
    }
    for (c = 0; c < m; c++)
        for (d = 0; d < n; d++)
            transpose[d][c] = matrix[c][d];
    printf("Transpose of the given matrix is\n");
    for(c=0;c<n;c++)
    {
        for(d=0;d<m;d++)
        {
            printf("%d ", transpose[c][d]);
        }
        printf("\n");
    }
    if(m==n)/*check if order is same */
    {
        for (c=0;c<m;c++)
        {
            for(d=0;d<n;d++)
            {
                if(matrix[c][d]!=transpose[c][d])
                    break;
            }
            if(d!=m)
                break;
        }
        if(c==m)
            printf("The given matrix is symmetric matrix\n");
        else
            printf("The given matrix is not symmetric matrix\n");
    }
    else
        printf("The given matrix is not symmetric matrix\n");
    return 0;
}

```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter the order of matrix :

2 2

Enter 4 elements :

1 2 3 4

The given matrix is

1 2

3 4

Transpose of the given matrix is

1 3

2 4

The given matrix is not symmetric matrix

ID: 22K61A05B4 Page No: 61

Test Case - 2

User Output

Enter the order of matrix :

2 2

Enter 4 elements :

4 5 5 4

The given matrix is

4 5

5 4

Transpose of the given matrix is

4 5

5 4

The given matrix is symmetric matrix

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 3

User Output

Enter the order of matrix :

3 2

Enter 6 elements :

1 2 3 4 5 6

The given matrix is

1 2

3 4

5 6

Transpose of the given matrix is

1 3 5

2 4 6

The given matrix is not symmetric matrix

Test Case - 4

User Output

```
Enter the order of matrix :  
3 3  
Enter 9 elements :  
1 1 1 1 1 1 1 1 1  
The given matrix is  
1 1 1  
1 1 1  
1 1 1  
Transpose of the given matrix is  
1 1 1  
1 1 1  
1 1 1  
The given matrix is symmetric matrix
```

ID: 22K61A05B4 Page No: 62

Sasi Institute of Technology and Engineering (Autonomous)
2022-2026-CSE-B

S.No: 26

Exp. Name: **Write a C program to implement String manipulation operations using library functions**

Date: 2023-01-19

Aim:

Write a program to implement the **string manipulation operations** by using string library functions.

At the time of execution, the program should print the message on the console as:

Enter two strings :

For example, if the user gives the **input** as:

Enter two strings : Ram Laxman

then the program should **print** the result as:

The length of Ram : 3
The copied string of Ram : Ram
Ram is greater than Laxman
The concatenated string : RamLaxman

Note: Do use the **printf()** function with a **newline** character (`\n`) at the end.

Source Code:

Program612.c

```
#include<stdio.h>
#include<string.h>
void main()
{
    char str1[20],str2[20],copy[20];
    int i;
    printf("Enter two strings : ");
    scanf("%s",str1);
    scanf("%s",str2);
    i=strlen(str1);
    printf("The length of %s : %d\n",str1,i);
    strcpy(copy,str1);
    printf("The copied string of %s : %s\n",str1,copy);
    i=strcmp(str1,str2);
    if(i==0)
    {
        printf("Both strings are equal\n");
    }
    else if(i<0)
    {
        printf("%s is less than %s\n",str1,str2);
    }
    else
    {
        printf("%s is greater than %s\n",str1,str2);
    }
    strcat(str1,str2);
    printf("The concatenated string : %s\n",str1);
}
```

ID: 22K61A05B4 Page No: 63

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
Enter two strings :  
Ram Laxman  
The length of Ram : 3  
The copied string of Ram : Ram  
Ram is greater than Laxman  
The concatenated string : RamLaxman
```

ID: 22K61A05B4 Page No: 64

Test Case - 2

User Output

```
Enter two strings :  
Teacher Student  
The length of Teacher : 7  
The copied string of Teacher : Teacher  
Teacher is greater than Student  
The concatenated string : TeacherStudent
```

2022-2026-CSE-B

Test Case - 3

User Output

```
Enter two strings :  
Languages Programming  
The length of Languages : 9  
The copied string of Languages : Languages  
Languages is less than Programming  
The concatenated string : LanguagesProgramming
```

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 4

User Output

```
Enter two strings :  
Ganga Ganga  
The length of Ganga : 5  
The copied string of Ganga : Ganga  
Both strings are equal  
The concatenated string : GangaGanga
```

S.No: 27

Exp. Name: **Write a C program to find the Length of a given string without using string library functions**

Date: 2023-01-19

Aim:

Write a program to find the **length** of a given string without using string library functions.

At the time of execution, the program should print the message on the console as:

Enter a string :

For example, if the user gives the **input** as:

Enter a string : Ganga

then the program should **print** the result as:

Length 5

Note: Do use the **printf()** function with a **newline** character (**\n**) at the end.

Source Code:

Program601.c

```
#include<stdio.h>
int main()
{
    char s[20];
    int i;
    printf("Enter a string : ");
    scanf("%s",s);
    for(i=0;s[i]!='\0';i++);
    printf("length %d\n",i);
    return 0;
}
```

ID: 22K61A05B4 Page No: 65

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter a string :

Amaravathi

length 10

S.No: 28

Exp. Name: **Write a C program to Copy one string into another string without using string library functions**

Date: 2023-01-19

Aim:

Write a program to [copy](#) a given string into another string without using string library functions.

At the time of execution, the program should print the message on the console as:

Enter a string :

For example, if the user gives the [input](#) as:

Enter a string : Narmada

then the program should [print](#) the result as:

The copied string = Narmada

Note: Do use the [printf\(\)](#) function with a [newline](#) character ([\n](#)) at the end.

Source Code:

Program604.c

```
#include<stdio.h>
int main()
{
    char s1[100],s2[100],i;
    printf("Enter a string : ");
    scanf("%s",s1);
    for(i=0;s1[i]!='\0';i++)
    {
        s2[i]=s1[i];
    }
    s2[i]='\0';
    printf("The copied string = %s\n",s2);
    return 0;
}
```

ID: 22K61A05B4 Page No: 66

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter a string :

Hyderabad

The copied string = Hyderabad

Test Case - 2

User Output

Enter a string :

London_UK

The copied string = London_UK

S.No: 29

Exp. Name: **Write a C program to Concatenate two given strings without using string library functions**

Date: 2023-01-19

Aim:

Write a program to [concatenate](#) two given strings without using string library functions.

At the time of execution, the program should print the message on the console as:

string1 :

For example, if the user gives the **input** as:

string1 : ILove

Next, the program should print the message on the console as:

string2 :

For example, if the user gives the **input** as:

string2 : Coding

then the program should **print** the result as:

concatenated string = ILoveCoding

Note: Do use the **printf()** function with a **newline** character ([\n](#)) at the end.

Source Code:

Program605.c

```
#include<stdio.h>
int main()
{
    char s1[100],s2[100],s3[200],i,j;
    printf("string1 : ");
    scanf("%s",s1);
    printf("string2 : ");
    scanf("%s",s2);
    for(i=0;s1[i]!='\0';i++)
    {
        s3[i] = s1[i];
    }
    j=i;
    for(i=0;s2[i]!='\0';i++)
    {
        s3[j] = s2[i];
        j++;
    }
    s3[j]='\0';
    printf("concatenated string = %s\n",s3);
    return 0;
}
```

ID: 22K61A05B4 Page No: 68

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
string1 :
I Love
string2 :
Coding
concatenated string = ILoveCoding

Test Case - 2
User Output
string1 :
1234
string2 :
567
concatenated string = 1234567

ID: 22K61A05B4 Page No: 69

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 30	Exp. Name: <i>Write a C program to check whether two given strings are Equal or not without using string library functions</i>	Date: 2023-01-19
----------	---	------------------

Aim:

Write a program to check whether the given two strings are **equal** or not without using string library functions.

At the time of execution, the program should print the message on the console as:

string1 :

For example, if the user gives the **input** as:

string1 : Kaveri

Next, the program should print the message on the console as:

string2 :

For example, if the user gives the **input** as:

string2 : Kaveri

then the program should **print** the result as:

equal

In the same way if the input is given as "Kaveri" and "kaveri", then the result will be "**not equal**".

Note: Do use the **printf()** function with a **newline** character (`\n`) at the end.

Source Code:

Program606.c

```

#include<stdio.h>
#include<string.h>
int main()
{
    char str1[100],str2[100];
    int result=0,i;
    printf("string1 : ");
    scanf("%s",str1);
    printf("string2 : ");
    scanf("%s",str2);
    while(str1[i]!='\0' && str2[i]!='\0')
    {
        if(str1[i]!=str2[i])
        {
            result=1;
            break;
        }
        i++;
    }
    if(result==0)
    printf("equal\n");
    else
    printf("not equal\n");
    return 0;
}

```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

string1 :
Kaveri
string2 :
Kaveri
equal

Test Case - 2

User Output

string1 :
ganga
string2 :
gangga
not equal

S.No: 31	Exp. Name: <i>Write a C program to demonstrate Functions with arguments and with return value - Program to perform Arithmetic Operations using Functions</i>	Date: 2023-01-19
----------	---	------------------

Aim:

Write a program to find the **addition**, **subtraction**, **multiplication** and **division** of two integer variables using **functions**.

At the time of execution, the program should print the message on the console as:

Enter two values :

For example, if the user gives the **input** as:

Enter two values : 23 11

then the program should **print** the result as:

Addition of two values = 34
 Subtraction of two values = 12
 multiplication of two values = 253
 division of two values = 2

Note: Write the functions in [Program701a.c](#).

Source Code:

Program701.c

```
#include <stdio.h>
#include "Program701a.c"
void main() {
    int a, b;
    printf("Enter two values : ");
    scanf("%d%d", &a, &b);
    printf("Addition of two values = %d\n", sum(a, b));
    printf("Subtraction of two values = %d\n", sub(a, b));
    printf("multiplication of two values = %d\n", mul(a, b));
    printf("division of two values = %d\n", div(a, b));
}
```

Program701a.c

ID: 22K61A05B4 Page No: 72

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include<stdio.h>
int sum(int a,int b);
int sub(int a,int b);
int mul(int a,int b);
int div(int a,int b);
int sum(int a,int b)
{
    return(a+b);
}
int sub(int a,int b)
{
    return(a-b);
}
int mul(int a,int b)
{
    return(a*b);
}
int div(int a,int b)
{
    return(a/b);
}

```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```

Enter two values :
23 11
Addition of two values = 34
Subtraction of two values = 12
multiplication of two values = 253
division of two values = 2

```

Test Case - 2

User Output

```

Enter two values :
55 5
Addition of two values = 60
Subtraction of two values = 50
multiplication of two values = 275
division of two values = 11

```

S.No: 32

Exp. Name: **Write a C program to demonstrate Functions with arguments and without return value**

Date: 2023-01-19

Aim:

Write a program to find the **largest** of given two numbers using **functions**.

At the time of execution, the program should print the message on the console as:

Enter two numbers :

For example, if the user gives the **input** as:

Enter two numbers : 22 55

then the program should **print** the result as:

Largest element = 55

Note: Write the function **largest()** in [program702a.c](#).

Source Code:

Program702.c

```
#include <stdio.h>
#include "Program702a.c"
void main() {
    int a, b;
    printf("Enter two numbers : ");
    scanf("%d%d" , &a, &b);
    largest(a, b);
}
```

ID: 22K61A05B4 Page No: 74

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Program702a.c

```
#include<stdio.h>
void largest(int a,int b);
void largest(int a,int b)
{
    if(a>b)
        printf("Largest element = %d\n",a);
    else
        printf("Largest element = %d\n",b);
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter two numbers :

22 55

Largest element = 55

Test Case - 2

User Output

Enter two numbers :

48 24

Largest element = 48

ID: 22K61A05B4 Page No: 75

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 33

Exp. Name: **Write a C program to demonstrate Functions without arguments and without return value**

Date: 2023-01-19

Aim:

Write a C program to demonstrate functions without arguments and without return value.

Write the functions **print()** and **hello()**.

The output is:

```
....***...
Hello! CodeTantra
....***...
```

Source Code:

FunctionCategories2.c

```
#include <stdio.h>
// Write the functions

void main() {
    print();
    hello();
    print();
}
print()
{
    printf("....***...\n");
}
hello()
{
    printf("Hello! CodeTantra\n");
}
```

ID: 22K61A05B4 Page No: 76

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
....***...
Hello! CodeTantra
....***...
```

S.No: 34

Exp. Name: **Write a C program to demonstrate Functions without arguments and with return value**

Date: 2023-01-19

Aim:

Write a C program to demonstrate functions without arguments and with return value.

The below code is used to check whether the given number is a prime number or not.

Write the function **prime()**.

Sample Input and Output:

```
Enter a number : 5
The given number is a prime number
```

ID: 22K61A05B4 Page No: 77

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Source Code:

FunctionCategories8.c

```
#include <stdio.h>
int prime();
void main() {
    if (prime() == 0) {
        printf("The given number is a prime number\n");
    } else {
        printf("The given number is not a prime number\n");
    }
}
// Write the function prime()
int prime()
{
int i,n,count=0;
printf("Enter a number : ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
    if(n%i==0)
        count++;
}
if(count==2)
return 0;
else
return count;
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter a number :
5

The given number is a prime number

Test Case - 2

User Output

Enter a number :

27

The given number is not a prime number

ID: 22K61A05B4 Page No: 78

Test Case - 3

User Output

Enter a number :

121

The given number is not a prime number

2022-2026-CSE-B

Test Case - 4

User Output

Enter a number :

1

The given number is not a prime number

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 5

User Output

Enter a number :

117

The given number is not a prime number

Test Case - 6

User Output

Enter a number :

137

The given number is a prime number

S.No: 35

Exp. Name: **Write a C program to Swap two values by using Call-by-Address method**

Date: 2023-01-19

Aim:

Write a program to **swap** two values by using **call by address** method.

At the time of execution, the program should print the message on the console as:

Enter two integer values :

For example, if the user gives the **input** as:

Enter two integer values : 12 13

then the program should **print** the result as:

Before swapping in main : a = 12 b = 13
After swapping in swap : *p = 13 *q = 12
After swapping in main : a = 13 b = 12

Note: Write the function **swap()** in [Program1002a.c](#) and do use the **printf()** function with a **newline** character ([\n](#)).

Source Code:

[Program1002.c](#)

```
#include <stdio.h>
#include "Program1002a.c"
void main() {
    int a, b;
    printf("Enter two integer values : ");
    scanf("%d %d", &a, &b);
    printf("Before swapping in main : a = %d b = %d\n", a, b);
    swap(&a, &b);
    printf("After swapping in main : a = %d b = %d\n", a, b);
}
```

[Program1002a.c](#)

```
#include<stdio.h>
void swap(int*p,int*q);
void swap(int*p,int*q)
{
    int temp;
    temp=*p;
    *p=*q;
    *q=temp;
    printf("After swapping in swap : *p = %d *q = %d\n",*p,*q);
}
```

ID: 22K61A05B4 Page No: 79

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
Enter two integer values :  
121 131  
Before swapping in main : a = 121 b = 131  
After swapping in swap : *p = 131 *q = 121  
After swapping in main : a = 131 b = 121
```

Page No: 80

ID: 22K61A05B4

Test Case - 2

User Output

```
Enter two integer values :  
555 999  
Before swapping in main : a = 555 b = 999  
After swapping in swap : *p = 999 *q = 555  
After swapping in main : a = 999 b = 555
```

2022-2026-CSE-B

Test Case - 3

User Output

```
Enter two integer values :  
1001 101  
Before swapping in main : a = 1001 b = 101  
After swapping in swap : *p = 101 *q = 1001  
After swapping in main : a = 101 b = 1001
```

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 4

User Output

```
Enter two integer values :  
9999 2999  
Before swapping in main : a = 9999 b = 2999  
After swapping in swap : *p = 2999 *q = 9999  
After swapping in main : a = 2999 b = 9999
```

Test Case - 5

User Output

```
Enter two integer values :  
10101 11010  
Before swapping in main : a = 10101 b = 11010  
After swapping in swap : *p = 11010 *q = 10101  
After swapping in main : a = 11010 b = 10101
```

S.No: 36

Exp. Name: **Write a C program to find the Factorial of a given number using Functions**

Date: 2023-01-19

Aim:

Write a program to find the **factorial** of a given number using **functions**.

At the time of execution, the program should print the message on the console as:

Enter a number :

For example, if the user gives the **input** as:

Enter a number : 5

then the program should **print** the result as:

Factorial of a given number 5 = 120

Note: Write the function **factorial()** in [Program704a.c](#).

Source Code:

Program704.c

```
#include <stdio.h>
#include "Program704a.c"
void main() {
    int number;
    printf("Enter a number : ");
    scanf("%d", &number);
    printf("Factorial of a given number %d = %d\n", number, factorial(number));
}
```

Program704a.c

```
#include<stdio.h>
int factorial(int n);
int factorial(int n)
{
    int i,fact=1;
    for(i=1;i<=n;i++)
        fact=fact*i;
    return fact;
}
```

ID: 22K61A05B4 Page No: 81

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter a number :

3

Factorial of a given number 3 = 6

Test Case - 2

User Output

Enter a number :

6

Factorial of a given number 6 = 720

ID: 22K61A05B4 Page No: 82

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 37

Exp. Name: **Write a C program to find the Factorial of a given number using Recursion**

Date: 2023-01-19

Aim:

Write a program to find the **factorial** of a given number using recursion process.

At the time of execution, the program should print the message on the console as:

Enter an integer :

For example, if the user gives the **input** as:

Enter an integer : 6

then the program should **print** the result as:

Factorial of 6 is : 720

Note: Write the recursive function **factorial()** in [Program901a.c](#).

Source Code:

Program901.c

```
#include <stdio.h>
#include "Program901a.c"
void main() {
    long int n;
    printf("Enter an integer : ");
    scanf("%ld", &n);
    printf("Factorial of %ld is : %ld\n", n ,factorial(n));
}
```

Program901a.c

```
#include<stdio.h>
long int factorial(long int n);
long int factorial(long int n)
{
    if(n==0)
        return 1;
    else
        return(n*factorial(n-1));
}
```

ID: 22K61A05B4 Page No: 83

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter an integer :

5

Factorial of 5 is : 120

Test Case - 2

User Output

Enter an integer :

7

Factorial of 7 is : 5040

ID: 22K61A05B4 Page No: 84

Test Case - 3

User Output

Enter an integer :

4

Factorial of 4 is : 24

2022-2026-CSE-B

Test Case - 4

User Output

Enter an integer :

8

Factorial of 8 is : 40320

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 5

User Output

Enter an integer :

0

Factorial of 0 is : 1

Test Case - 6

User Output

Enter an integer :

9

Factorial of 9 is : 362880

S.No: 38

Exp. Name: **Write a program to find GCD of two numbers using Functions**

Date: 2023-01-20

Aim:

Write a C program to find [GCD](#) of two integer numbers using functions.

At the time of execution, the program should print the message on the console as:

Enter two integer numbers :

For example, if the user gives the **input** as :

Enter two integer numbers : 144 428

then the program should **print** the result as:

GCD of 144 and 428 : 4

Note: Write the function **gcdResult()** in [FunctionsExample3a.c](#).

Source Code:

FunctionsExample3.c

```
#include <stdio.h>
#include "FunctionsExample3a.c"
void main() {
    int num1, num2;
    printf("Enter two integer numbers : ");
    scanf("%d%d", &num1, &num2);
    printf("GCD of %d and %d : %d\n", num1, num2, gcdResult(num1, num2));
}
```

FunctionsExample3a.c

```
int gcdResult(int n1,int n2);
int gcdResult(int n1,int n2)
{
    int i,gcd;
    for(i=1;i<=n2;i++)
    {
        if(n1%i==0&&n2%i==0)
            gcd=i;
    }
    return gcd;
}
```

ID: 22K61A05B4 Page No: 85

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter two integer numbers :

144 428

GCD of 144 and 428 : 4

Test Case - 2

User Output

Enter two integer numbers :

27 113

GCD of 27 and 113 : 1

ID: 22K61A05B4 Page No: 86

Test Case - 3

User Output

Enter two integer numbers :

55 100

GCD of 55 and 100 : 5

2022-2026-CSE-B

Test Case - 4

User Output

Enter two integer numbers :

125 175

GCD of 125 and 175 : 25

Sasi Institute of Technology and Engineering (Autonomous)

Test Case - 5

User Output

Enter two integer numbers :

124 156

GCD of 124 and 156 : 4

S.No: 39

Exp. Name: **Write a C program to find the GCD of two numbers using Recursion**

Date: 2023-01-23

Aim:

Write a program to find the **gcd** (Greatest Common Divisor) of a given two numbers using recursion process.

The greatest common divisor (**gcd**) of two or more integers, when at least one of them is not zero, is the largest positive integer that is a divisor of both numbers.

At the time of execution, the program should print the message on the console as:

Enter two integer values :

For example, if the user gives the **input** as:

Enter two integer values : 12 18

then the program should **print** the result as:

The gcd of two numbers 12 and 18 = 6

Note: Write the recursive function **gcd()** in [Program906a.c](#).

Source Code:

Program906.c

```
#include <stdio.h>
#include "Program906a.c"
void main() {
    int a, b;
    printf("Enter two integer values : ");
    scanf("%d %d", &a, &b);
    printf("The gcd of two numbers %d and %d = %d\n", a, b, gcd(a, b));
}
```

Program906a.c

```
#include<stdio.h>
int gcd(int n1,int n2);
int gcd(int n1,int n2)
{
    if(n2!=0)
        return gcd(n2,n1%n2);
    else
        return n1;
}
```

ID: 22K61A05B4 Page No: 87

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter two integer values :

12 15

The gcd of two numbers 12 and 15 = 3

Test Case - 2**User Output**

Enter two integer values :

36 124

The gcd of two numbers 36 and 124 = 4

Page No: 88

ID: 22K61A05B4
2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Aim:

Write a C program to calculate x^n using functions.

Sample Input and Output:

```
Enter value of x : 1.5
Enter value of  : 2
1.500000^2 = 2.250000
```

Note-1: Let us consider x as real number and n as integer number.

Note-2: Write the function **power()** in [FunctionsExample5a.c](#).

Source Code:**FunctionsExample5.c**

```
#include <stdio.h>
#include "FunctionsExample5a.c"
void main() {
    float result, x;
    int n;
    printf("Enter value of x : ");
    scanf("%f", &x);
    printf("Enter value of  : ");
    scanf("%d", &n);
    result = power(x, n);
    printf("%f^%d = %f\n", x, n, result);
}
```

FunctionsExample5a.c

```
#include<stdio.h>
float power(float x,int y);
float power(float x,int y)
{
    int i;
    float result=x;
    for(i=1;i<y;i++)
        result=result*x;
    return result;
}
```

Execution Results - All test cases have succeeded!**Test Case - 1****User Output**

Enter value of x :

1.5

Enter value of :
2
$1.500000^2 = 2.250000$

Test Case - 2
User Output
Enter value of x :
3.57
Enter value of :
3
$3.570000^3 = 45.499290$

Test Case - 3
User Output
Enter value of x :
25.75
Enter value of :
3
$25.750000^3 = 17073.859375$

S.No: 41

Exp. Name: **Write a C Program to find the Power of a given number using Recursion**

Date: 2023-01-19

Aim:

Write a program to find the **power** of a given number using recursion process.

At the time of execution, the program should print the message on the console as:

Enter a number :

For example, if the user gives the **input** as:

Enter a number : 5

Next, the program should print the message on the console as:

Enter power :

For example, if the user gives the **input** as:

Enter power : 4

then the program should **print** the result as:

5 to the power of 4 is : 625

Note: Write the recursive function **power()** in [Program905a.c](#).

Source Code:

[Program905.c](#)

```
#include <stdio.h>
#include "Program905a.c"
void main() {
    int m, n;
    printf("Enter a number : ");
    scanf("%d", &m);
    printf("Enter power : ");
    scanf("%d", &n);
    printf("%d to the power of %d is : %d\n", m, n, power(m, n));
}
```

[Program905a.c](#)

```
#include<stdio.h>
int power(int x,int y);
int power(int x,int y)
{
    if(y==0)
        return 1;
    else
        return(x*power(x,y-1));
}
```

ID: 22K61A05B4 Page No: 91

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter a number :
3
Enter power :
4
3 to the power of 4 is : 81

Test Case - 2
User Output
Enter a number :
4
Enter power :
3
4 to the power of 3 is : 64

ID: 22K61A05B4 Page No: 92

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 42

Exp. Name: **Write a C program to print the Fibonacci series up to the given limit using Functions**

Date: 2023-01-19

Aim:

Write a program to print the [Fibonacci series](#) up to the given limit.

At the time of execution, the program should print the message on the console as:

Enter the maximum limit to generate the Fibonacci series :

For example, if the user gives the **input** as:

Enter the maximum limit to generate the Fibonacci series : 15

then the program should **print** the result as:

The Fibonacci series is : 0 1 1 2 3 5 8 13

Note: Write the function **fibonacci()** in [Program708a.c](#).

Source Code:

Program708.c

```
#include <stdio.h>
#include "Program708a.c"
void main() {
    int number;
    printf("Enter the maximum limit to generate the Fibonacci series : ");
    scanf("%d", &number);
    fibonacci(number);
}
```

ID: 22K61A05B4 Page No: 93

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Program708a.c

```
#include<stdio.h>
void fibonacci(int num)
{
    int i,fib1=0,fib2=1,fib3;
    printf("The Fibonacci series is : ");
    printf("%i %i",fib1,fib2);
    fib3=fib1+fib2;
    while(fib3<num)
    {
        printf(" %i", fib3);
        fib1=fib2;
        fib2=fib3;
        fib3=fib1+fib2;
    }
    printf("\n");
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter the maximum limit to generate the Fibonacci series :

30

The Fibonacci series is : 0 1 1 2 3 5 8 13 21

S.No: 43

Exp. Name: **Write a C program to display the Fibonacci series up to the given number of terms using Recursion**

Date: 2023-01-19

Aim:

Write a program to display the [fibonacci series](#) up to the given number of terms using recursion process.

The fibonacci series is [0 1 1 2 3 5 8 13 21 34.....](#).

At the time of execution, the program should print the message on the console as:

Enter value of n :

For example, if the user gives the **input** as:

Enter value of n : 6

then the program should **print** the result as:

The fibonacci series of 6 terms are : 0 1 1 2 3 5

Note: Write the recursive function **fib()** in [Program908a.c](#).

Source Code:

[Program908.c](#)

```
#include <stdio.h>
#include "Program908a.c"
void main() {
    int n, i;
    printf("Enter value of n : ");
    scanf("%d", &n);
    printf("The fibonacci series of %d terms are : ", n);
    for (i = 0; i < n; i++) {
        printf(" %d ", fib(i));
    }
}
```

ID: 22K61A05B4 Page No: 95

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

[Program908a.c](#)

```
#include<stdio.h>
int fib(int n);
int fib(int n)
{
    if(n==0)
        return 0;
    else if(n==1)
        return 1;
    else
        return(fib(n-1)+fib(n-2));
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter value of n :

4

The fibonacci series of 4 terms are : 0 1 1 2

Test Case - 2

User Output

Enter value of n :

8

The fibonacci series of 8 terms are : 0 1 1 2 3 5 8 13

ID: 22K61A05B4 Page No: 96

Test Case - 3

User Output

Enter value of n :

14

The fibonacci series of 14 terms are : 0 1 1 2 3 5 8 13 21 34 55 89 144 233

2022-2026-CSE-B

Test Case - 4

User Output

Enter value of n :

3

The fibonacci series of 3 terms are : 0 1 1

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 44

Exp. Name: **Write a C program to find Sum of array elements by allocating memory using malloc() function**

Date: 2023-01-19

Aim:

Write a program to find the **sum** of n elements by allocating memory by using **malloc()** function.

At the time of execution, the program should print the message on the console as:

Enter n value :

For example, if the user gives the **input** as:

Enter n value : 4

Next, the program should print the message on the console as:

Enter 4 values :

For example, if the user gives the **input** as:

Enter 4 values : 1 5 4 2

then the program should **print** the result as:

The sum of given array elements : 12

Note: Write the functions **allocateMemory()**, **read()** and **sum()** in [UsingMalloc.c](#).

Source Code:

SumOfArray1.c

```
#include <stdio.h>
#include <stdlib.h>
#include "UsingMalloc.c"
void main() {
    int *p, n, i;
    printf("Enter n value : ");
    scanf("%d", &n);
    p = allocateMemory(n);
    printf("Enter %d values : ", n);
    read(p, n);
    printf("The sum of given array elements : %d\n", sum(p, n));
}
```

UsingMalloc.c

ID: 22K61A05B4 Page No: 97

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include<stdio.h>
#include<stdlib.h>
int *allocateMemory(int n);
void read(int*,int);
int sum(int*,int);
int*allocateMemory(int n)
{
    int*p;
    p=(int*)malloc(n*sizeof(int));
    return p;
}
void read(int*p,int n)
{
    int i,x;
    for(i=0;i<n;i++)
    {
        scanf("%d",p);
        p++;
    }
}
int sum(int*p,int n)
{
    int i,sum=0;
    for(i=0;i<n;i++)
    sum=sum+p[i];
    return sum;
}

```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter n value :

4

Enter 4 values :

1 4 5 2

The sum of given array elements : 12

Test Case - 2

User Output

Enter n value :

3

Enter 3 values :

10 20 30

The sum of given array elements : 60

Test Case - 3

User Output

Enter n value :

4

Enter 4 values :

-5 -6 -4 -2

The sum of given array elements : -17

S.No: 45

Exp. Name: **Write a C program to find Sum of array elements by allocating memory using calloc() function**

Date: 2023-01-19

Aim:

Write a program to find the **sum** of n elements by allocating memory by using **calloc()** function.

At the time of execution, the program should print the message on the console as:

Enter n value :

For example, if the user gives the **input** as:

Enter n value : 4

Next, the program should print the message on the console as:

Enter 4 values :

For example, if the user gives the **input** as:

Enter 4 values : 3 5 4 7

then the program should **print** the result as:

The sum of given array elements : 19

Note: Write the functions **allocateMemory()**, **read()** and **sum()** in [UsingCalloc.c](#).

Source Code:

SumOfArray2.c

```
#include <stdio.h>
#include <stdlib.h>
#include "UsingCalloc.c"
void main() {
    int *p, n, i;
    printf("Enter n value : ");
    scanf("%d", &n);
    p = allocateMemory(n);
    printf("Enter %d values : ", n);
    read(p, n);
    printf("The sum of given array elements : %d\n", sum(p, n));
}
```

UsingCalloc.c

ID: 2K61A05B4 Page No: 100

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include<stdio.h>
#include<stdlib.h>
int*allocateMemory(int n);
void read(int*,int);
int sum(int*,int);
int*allocateMemory(int n)
{
    int*p;
    p=(int*)calloc(n,sizeof(int));
    return p;
}
void read(int*p,int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        scanf("%d",p);
        p++;
    }
}
int sum(int*p,int n)
{
    int i,sum=0;
    for(i=0;i<n;i++)
    sum=sum+p[i];
    return sum;
}

```

ID: 22K61A05B4 Page No: 101

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter n value :
5
Enter 5 values :
2 5 33 11 26
The sum of given array elements : 77

S.No: 46

Exp. Name: **Write a C program to implement a Nested Structure to store and display the Student information**

Date: 2023-01-20

Aim:

Write a program to print student details using **nested structures**.

The structure `student` contains the fields regdno, name and date. `Date` is the nested structure and it contains the fields day, month and year.

At the time of execution, the program should print the following messages one by one on the console as:

```
Enter student number :  
Enter student name :  
Enter date of joining :
```

For example, if the user gives the **input** as:

```
Enter student number : 1002  
Enter student name : Saraswathi  
Enter date of joining : 31 01 2019
```

then the program should **print** the result as:

```
The student details are :  
Number : 1002  
Name : Saraswathi  
Date of joining : 31-1-2019
```

Note - 1: Do use the `printf()` function with a **newline** character (`\n`).

Note - 2: Write the functions `read()` and `display()` in [NestedStructureFunctions.c](#)

Source Code:

NestedStructureMain.c

```
#include <stdio.h>  
struct student {  
    int regdno;  
    char name[30];  
    struct date {  
        int day;  
        int month;  
        int year;  
    }doj;  
};  
#include "NestedStructureFunctions.c"  
void main() {  
    int i;  
    struct student s;  
    read(&s);  
    printf("The student details are : \n");  
    display(s);  
}
```

ID: 22K61A05B4 Page No: 102

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```
#include<stdio.h>
void read(struct student*s);
void display(struct student p);
void read(struct student*s)
{
    printf("Enter student number : ");
    scanf("%d",&s->regdno);
    printf("Enter student name : ");
    scanf("%s",s->name);
    printf("Enter date of joining : ");
    scanf("%d%d%d",&s->doj.day,&s->doj.month,&s->doj.year);
}
void display(struct student s)
{
    printf("Number : %d\n",s.regdno);
    printf("Name : %s\n",s.name);
    printf("Date of joining : %d-%d-%d\n",s.doj.day,s.doj.month,s.doj.year);
}
```

Execution Results - All test cases have succeeded!**Test Case - 1****User Output**

Enter student number :

1001

Enter student name :

Sachin

Enter date of joining :

12 06 1987

The student details are :

Number : 1001

Name : Sachin

Date of joining : 12-6-1987

Test Case - 2**User Output**

Enter student number :

2001

Enter student name :

Lakshmi

Enter date of joining :

15 08 2018

The student details are :

Number : 2001

Name : Lakshmi

Date of joining : 15-8-2018

S.No: 47

Exp. Name: **Write a C program to Open a File and to Print its contents on the screen**

Date: 2023-01-23

Aim:

Follow the instructions given below to write a program to [open](#) a file and to [print](#) its **contents** on the screen.

- Open a new file "[SampleText1.txt](#)" in write mode
- Write the content in the file
- Close the file
- Open the same file in read mode
- Read the content from file and print them on the screen
- Close the file

Source Code:

Program1501.c

```
#include <stdio.h>
void main() {
    FILE *fp;
    char ch;
    fp = fopen("SampleText1.txt", "w"); // Open a new file in write mode
    printf("Enter the text with @ at end : ");
    while ((ch=getchar())!='@') { // Repeat loop till read @ at the end
        putc(ch,fp); // Put read character onto the file
    }
    putc('@',fp); // Put delimiter @ at the end on the file
    fclose(fp); // Close the file
    fp = fopen("SampleText1.txt", "r"); // Open the existed file in read mode
    printf("Given message is : ");
    while ((ch=fgetc(fp))!='@') { // Repeat loop till get @ at the end of existed file
        putchar(ch); // Put the character on the screen
    }
    printf("\n");
    fclose(fp); // Close the file
}
```

ID: 22K61A05B4

Page No: 104

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter the text with @ at end :

CodeTantra is a

Startup Company recognized by Government
of India@

Given message is : CodeTantra is a

Startup Company recognized by Government of India

Test Case - 2

User Output

Enter the text with @ at end :

CodeTantra is
increasing development of Languages Year
by Year@
Given message is : CodeTantra is
increasing development of Languages Year
by Year

ID: 22K61A05B4 Page No: 105

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

S.No: 48

Exp. Name: **Write a C program to Copy contents of one File into another File**

Date: 2023-01-20

Aim:

Write a program to [copy](#) contents of one file into another file. Follow the instructions given below to write a program to copy the contents of one file to another file:

- Open a new file "[SampleTextFile1.txt](#)" in write mode
- Write the content onto the file
- Close the file
- Open an existing file "[SampleTextFile1.txt](#)" in read mode
- Open a new file "[SampleTextFile2.txt](#)" in write mode
- Copy the content from existing file to new file
- Close the files
- Open the copied file in read mode
- Read the text from file and print on the screen
- Close the file

ID: 22K61A05B4

Page No: 106

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Source Code:

Program1502.c

```
#include <stdio.h>
void main() {
    FILE *fp, *fp1, *fp2;
    char ch;
    fp = fopen("SampleTextFile1.txt", "w"); // Open a new file in write mode
    printf("Enter the text with @ at end : ");
    while ((ch=getchar())!='@') { // Read loop till get @ at the end
        putc(ch, fp);
    }
    putc(ch, fp);
    fclose(fp);
    fp1 = fopen("SampleTextFile1.txt", "r"); // Open an existed file in read mode
    fp2 = fopen("SampleTextFile2.txt", "w"); // Open a new file in write mode
    while ((ch=getc(fp1))!='@') { // Repeat loop till get @ at the end of existed file
        putc(ch, fp2);
    }
    putc(ch, fp2);
    fclose(fp1); // Close the existed file
    fclose(fp2); // Close the copied file
    fp2 = fopen("SampleTextFile2.txt", "r"); // Open the copied file in read mode
    printf("Copied text is : ");
    while ((ch=getc(fp2))!='@') { // Repeat loop till get @ at the end of copied file
        putchar(ch);
    }
    printf("\n");
    fclose(fp2); // Close the copied file
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

Enter the text with @ at end :

CodeTantra started in the year 2014@

Copied text is : CodeTantra started in the year 2014

Test Case - 2

User Output

Enter the text with @ at end :

CodeTantra received

best Startup award from Hysea in 2016@

Copied text is : CodeTantra received

best Startup award from Hysea in 2016

ID: 22K61A05B4 Page No: 107

Sasi Institute of Technology and Engineering (Autonomous) | 2022-2026-CSE-B

S.No: 49

Exp. Name: **Write a C program to Merge two Files and stores their contents in another File**

Date: 2023-01-23

Aim:

Write a program to `merge` two files and stores their contents in another file.

- Open a new file "`SampleDataFile1.txt`" in write mode
- Write the content onto the file
- Close the file
- Open another new file "`SampleDataFile2.txt`" in write mode
- Write the content onto the file
- Close the file
- Open first existing file "`SampleDataFile1.txt`" in read mode
- Open a new file "`SampleDataFile3.txt`" in write mode
- Copy the content from first existing file to new file
- Close the first existing file
- Open another existing file "`SampleDataFile2.txt`" in read mode
- Copy its content from existing file to new file
- Close that existing file
- Close the merged file

Source Code:

Program1503.c

ID: 22K61A05B4 Page No: 108

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

```

#include <stdio.h>
void main() {
    FILE *fp1, *fp2, *fp3;
    char ch;
    fp1 = fopen("SampleDataFile1.txt", "w"); // Open file in corresponding mode
    printf("Enter the text with @ at end for file-1 :\n");
    while ((ch=getchar())!='@') { // Write the condition
        putc(ch, fp1);
    }
    putc(ch, fp1);
    fclose(fp1);
    fp2 = fopen("SampleDataFile2.txt", "w"); // Open file in corresponding mode
    printf("Enter the text with @ at end for file-2 :\n");
    while ((ch=getchar())!='@') { // Write the condition
        putc(ch, fp2);
    }
    putc(ch, fp2);
    fclose(fp2);
    fp1 = fopen("SampleDataFile1.txt", "r"); // Open a first existed file in read mode
    fp3 = fopen("SampleDataFile3.txt", "w"); // Open a new file in write mode
    while ((ch=getc(fp1))!='@') { // Repeat loop till get @ at the end of existed file
        putc(ch, fp3);
    }
    fclose(fp1); // Close the first existed file
    fp2 = fopen("SampleDataFile2.txt", "r"); // Open a second existed file in read mode
    while ((ch=getc(fp2))!='@') { // Repeat loop till get @ at the end of existed file
        putc(ch, fp3);
    }
    putc(ch, fp3);
    fclose(fp2);
    fclose(fp3);
    fp3 = fopen("SampleDataFile3.txt", "r"); // Open the merged file in read mode
    printf("Merged text is : ");
    while ((ch=getc(fp3))!='@') { // Repeat loop till get @ at the end of merged file
        putchar(ch);
    }
    printf("\n");
    fclose(fp3); // Close the merged file
}

```

ID: 22K61A05B4 Page No: 109

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter the text with @ at end for file-1 :
CodeTantra@
Enter the text with @ at end for file-2 :
Interactive Tool@
Merged text is : CodeTantra
Interactive Tool

Test Case - 2

User Output

Enter the text with @ at end for file-1 :

CodeTantra developed an interactive tool

in the year 2014

CodeTantra got best Startup award in 2016@

Enter the text with @ at end for file-2 :

Now lot of Companies and Colleges using

CodeTantra Tool@

Merged text is : CodeTantra developed an interactive tool

in the year 2014

CodeTantra got best Startup award in 2016

Now lot of Companies and Colleges using CodeTantra Tool

ID: 22K61A05B4 Page No: 110

Sasi Institute of Technology and Engineering (Autonomous)

2022-2026-CSE-B

Aim:

Write a program to [delete](#) a **file**.

Note: Use the [remove\(fileName\)](#) function to delete an existing file.

Source Code:

Program1504.c

```
#include <stdio.h>
void main() {
    FILE *fp;
    int status;
    char fileName[40], ch;
    printf("Enter a new file name : ");
    gets(fileName);
    fp = fopen(fileName,"w"); // Open a new file in write mode
    printf("Enter the text with @ at end : ");
    while ((ch=getchar())!='@') { // Repeat loop till read @ at the end
        putc(ch,fp); // Put read character onto the file
    }
    putc(ch,fp); // Put delimiter @ at the end on the file
    fclose(fp); // Close the file
    fp = fopen(fileName,"r"); // Open the existed file in read mode
    printf("Given message is : ");
    while ((ch=getc(fp))!='@') { // Repeat loop till get @ at the end of existed file
        putchar(ch); // Put the character on the screen
    }
    printf("\n");
    fclose(fp); // Close the file
    status = remove(fileName); // Put the fileName to be removed
    if (status == 0)
        printf("%s file is deleted successfully\n",fileName); // Place the removed
    fileName
    else {
        printf("Unable to delete the file -- ");
        perror("Error\n");
    }
}
```

ID: 22K61A05B4

Page No: 111

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter a new file name :
Text1.txt
Enter the text with @ at end :
This is CodeTantra@
Given message is : This is CodeTantra
Text1.txt file is deleted successfully

Test Case - 2

User Output

Enter a new file name :

Text2.txt

Enter the text with @ at end :

C developed by Dennis Ritchie@

Given message is : C developed by Dennis Ritchie

Text2.txt file is deleted successfully

Page No: 112

ID: 22K61A05B4

2022-2026-CSE-B

Sasi Institute of Technology and Engineering (Autonomous)

