

## ACTIVITY 5:

Reg no:19BCS0012

### 1:GCD

```
#include<stdio.h>

void gcd();

void main()
{
    gcd();

    getch();
}

void gcd()
{
    int n1,i,n2,res;

    printf("enter the two number");

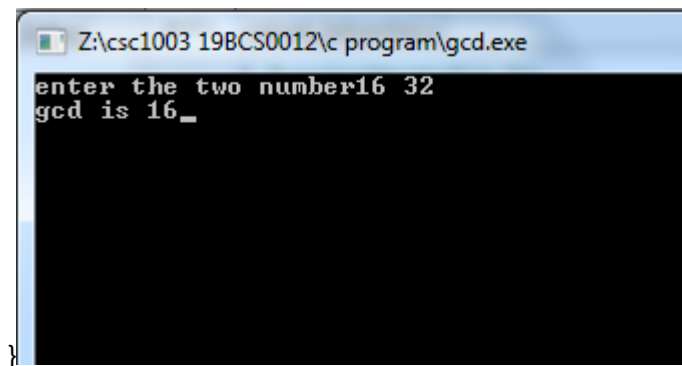
    scanf("%d%d",&n1,&n2);

    for(i=2;i<=n1&& n2;i++)
    {
        if(n1%i==0&& n2%i==0);

        res=i;
    }

    printf("gcd is %d",res);

}
```

A screenshot of a Windows command prompt window. The title bar shows the file path "Z:\csc1003 19BCS0012\c program\gcd.exe". The prompt displays the text "enter the two number" followed by the input "16 32". Below this, it shows the output "gcd is 16\_".

```
Z:\csc1003 19BCS0012\c program\gcd.exe
enter the two number16 32
gcd is 16_
```

## 2:NATURAL NUMBER FROM ONE TO N

```
#include<stdio.h>

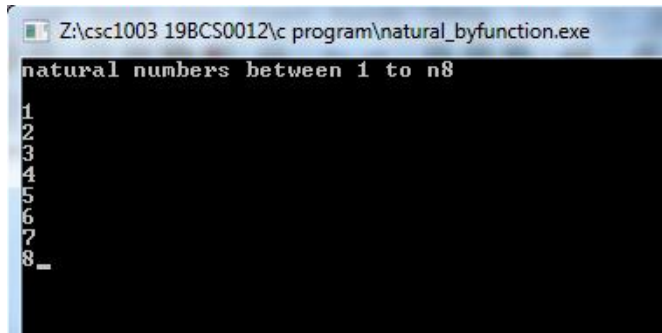
void natural();

void main()
{
    printf("natural numbers between 1 to n");
    natural();
    getch();
}

void natural()
{
    int i,n;
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        printf("\n%d",i);
    }
}
```

```
}
```

```
//OUTPUT
```



### 3.swap\_by\_call\_by\_value\_call\_by\_refference\_

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

```
void swap();
```

```
void main()
```

```
{
```

```
    swap();
```

```
    getch();
```

```
}
```

```
void swap()
```

```
{
```

```
    int temp,a,b;
```

```
    printf("enter the number two swap");
```

```
    scanf("%d%d",&a,&b);
```

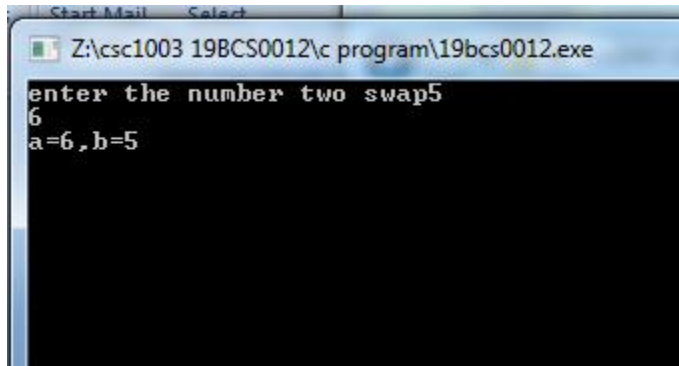
```
    temp=a;
```

```
    a=b;
```

```
    b=temp;
```

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

```
    getch();  
}
```



### Swaping using call by refrence

```
#include<stdio.h>  
  
void swap(int*,int*);  
  
void main()  
{  
    int a,b;  
    printf("enter the number two swap");  
    scanf("%d%d",&a,&b);  
    swap(&a,&b);  
    printf(" a=%d,b=%d",a,b);  
    getch();  
}  
  
void swap(int*a,int*b)
```

```

{
int temp;

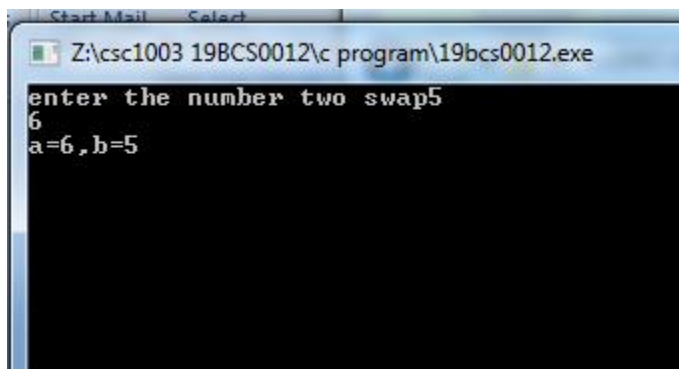
    temp=*a;

    *a=*b;

    *b=temp;

}

```



## 4.Factorial using the recursion:

```

#include<stdio.h>

#include<conio.h>

int fact(int);

void main()

{

    int x,n;

    printf(" Enter the Number to Find Factorial :");

    scanf("%d",&n);

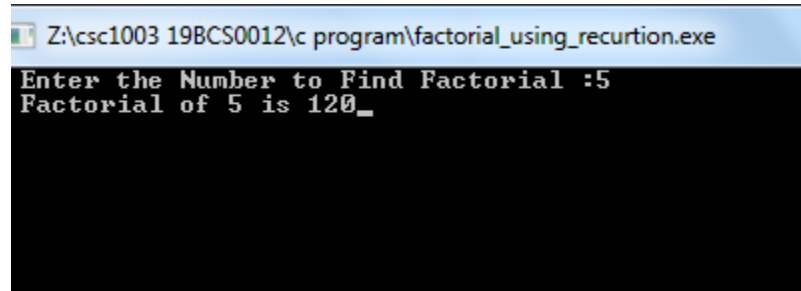
```

```
x=fact(n);

printf(" Factorial of %d is %d",n,x);


getch();
}

int fact(int n)
{
    if(n==0)
        return(1);
    return(n*fact(n-1));
}
```



Z:\csc1003 19BCS0012\c program\factorial\_using\_recurtion.exe

Enter the Number to Find Factorial :5  
Factorial of 5 is 120\_

5. Write a C program to calculate the following

i. sum:  $1 - x^2/2! + x^4/4! - x^6/6! + x^8/8! - x^{10}/10! + \dots$

```
void main()
{
    int counter,f_coun;
    float sum=0,x,power,fact;

    printf("\tEQUATION SERIES : 1- X^2/2! + X^4/4! - X^6/6! + X^8/8! - X^10/10!");

    printf("\n\tENTER VALUE OF X : ");
    scanf("%f",&x);

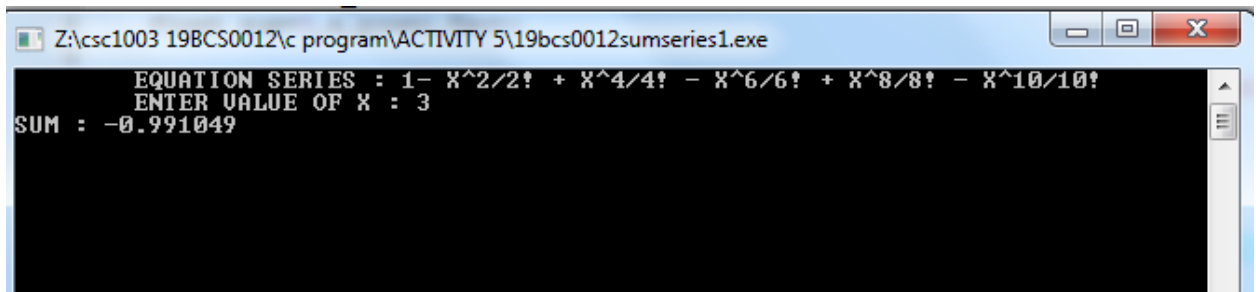
    for(counter=0, power=0; power<=10; counter++,power=power+2)
    {
        fact=1;

        for(f_coun=power; f_coun>=1; f_coun--)
            fact *= f_coun;

        sum=sum+(pow(-1,counter)*(pow(x,power)/fact));
    }

    printf("SUM : %f",sum);
    getch();
}
```

output



```
Z:\csc1003 19BCS0012\c program\ACTIVITY 5\19bcs0012sumseries1.exe
EQUATION SERIES : 1- X^2/2! + X^4/4! - X^6/6! + X^8/8! - X^10/10!
ENTER VALUE OF X : 3
SUM : -0.991049
```

ii) sum series

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

```
void main()
```

```
{
```

```
    float x,sum,no_row;
```

```
    int i,n;
```

```
    printf("Input the value of x :");
```

```
    scanf("%f",&x);
```

```
    printf("Input number of terms : ");
```

```
    scanf("%d",&n);
```

```
    sum =1; no_row = 1;
```

```
    for (i=1;i<n;i++)
```

```
    {
```

```
        no_row = no_row*x/(float)i;
```

```
        sum =sum+ no_row;
```

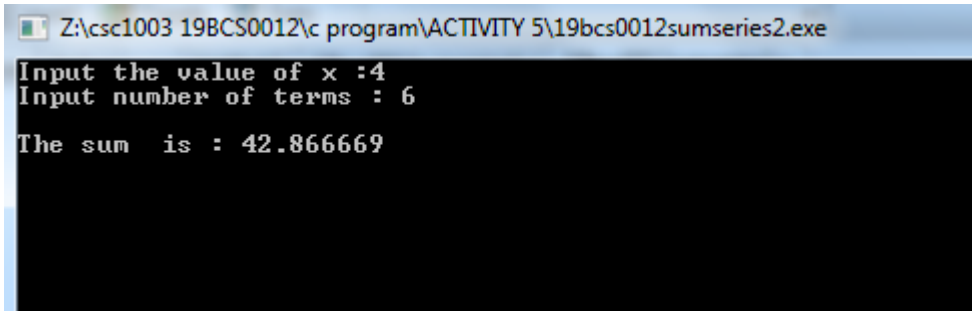
```
    }
```

```
    printf("\nThe sum is : %f\n",sum);
```

```
    getch();
```



output:



```
Z:\csc1003 19BCS0012\c program\ACTIVITY 5\19bcs0012sumseries2.exe
Input the value of x :4
Input number of terms : 6
The sum is : 42.866669
```

#### ACTIVITY 6

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

```
int main()
```

```
{
```

```
    int array[100], maximum, size, c, location = 1;
```

```
    printf("Enter the number of elements in array\n");
```

```
    scanf("%d", &size);
```

```
    printf("Enter %d integers\n", size);
```

```
    for (c = 0; c < size; c++)
```

```
        scanf("%d", &array[c]);
```

```
    maximum = array[0];
```

```
    for (c = 1; c < size; c++)
```

```
    {
```

```
        if (array[c] > maximum)
```

```

{
    maximum = array[c];
    location = c+1;
}
}

```

```

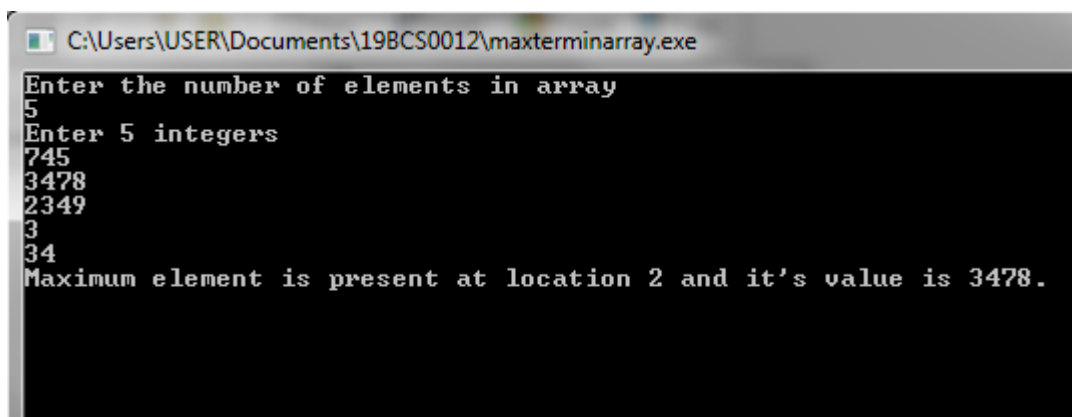
printf("Maximum element is present at location %d and it's value is %d.\n", location, maximum);

getch();

return 0;
}

```

OUTPUT



```

C:\Users\USER\Documents\19BCS0012\maxterminarray.exe
Enter the number of elements in array
5
Enter 5 integers
745
3478
2349
3
34
Maximum element is present at location 2 and it's value is 3478.

```

Find the count of total number of even and odd number from array

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

```
void main()
```

```
{
```

```
    int a[100],n,i,eve=0,odd=0;
```

```
    printf("Enter the number of elements in array\n");
```

```
scanf("%d", &n);
```

```
printf("Enter each integers\n");
```

```
for (i = 0; i < n; i++)
```

```
{
```

```
    scanf("%d", &a[i]);
```

```
}
```

```
for (i = 0; i < n; i++)
```

```
{
```

```
if(a[i]%2==0)
```

```
eve++;
```

```
else
```

```
odd++;
```

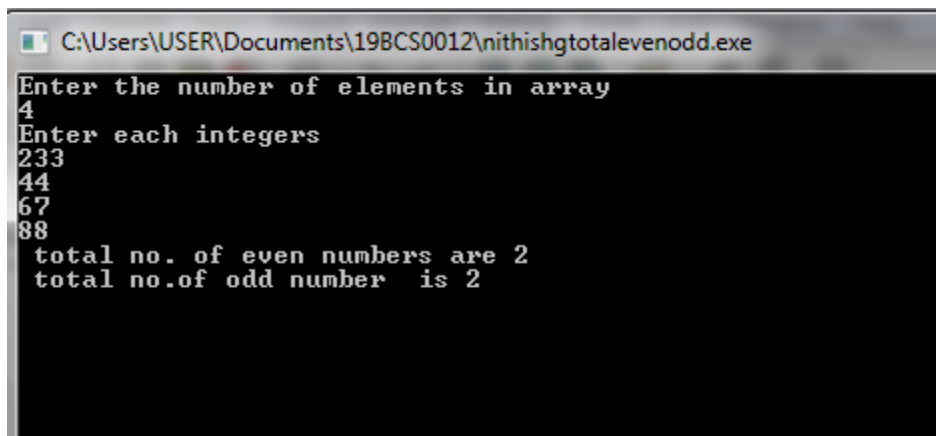
```
}
```

```
printf(" total no. of even numbers are %d\n total no.of odd number is %d",eve,odd);
```

```
getch();
```

```
}
```

Output



```
C:\Users\USER\Documents\19BCS0012\nithishgtotalevenodd.exe
Enter the number of elements in array
4
Enter each integers
233
44
67
88
total no. of even numbers are 2
total no.of odd number is 2
```

3.Search an element from array

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

```
#define MAX_SIZE 100 // Maximum array size
```

```
int main()
```

```
{
```

```
    int arr[MAX_SIZE];
```

```
    int size, i, toSearch, found;
```

```
    /* Input size of array */
```

```
    printf("Enter size of array: ");
```

```
    scanf("%d", &size);
```

```
    printf("Enter elements in array: ");
```

```
    for(i=0; i<size; i++)
```

```
    {
```

```
        scanf("%d", &arr[i]);
```

```
    }
```

```
printf("\nEnter element to search: ");

scanf("%d", &toSearch);

found = 0;


for(i=0; i<size; i++)
{

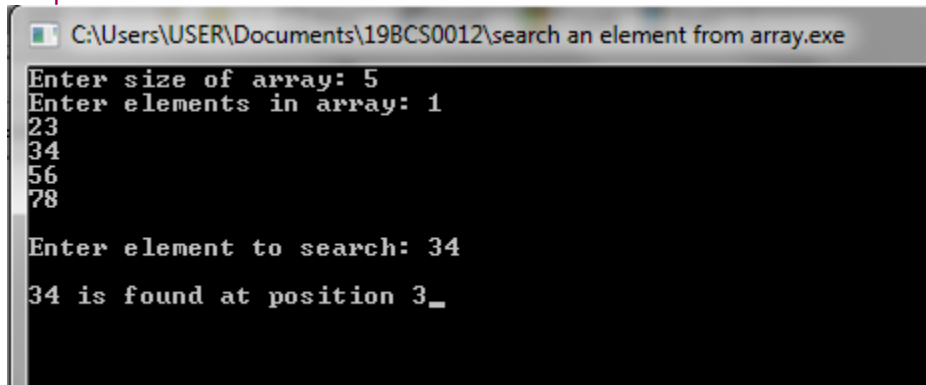
    if(arr[i] == toSearch)
    {
        found = 1;
        break;
    }
}


if(found == 1)
{
    printf("\n%d is found at position %d", toSearch, i + 1);
}
else
{
    printf("\n%d is not found in the array", toSearch);
}

getch();

return 0;
}
```

## Output



```
C:\Users\USER\Documents\19BCS0012\search an element from array.exe
Enter size of array: 5
Enter elements in array: 1
23
34
56
78

Enter element to search: 34
34 is found at position 3_
```

## 4.Sort the array in ascending order

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

    int i, j, a, n, number[30];
    printf("Enter the value of N \n");
    scanf("%d", &n);

    printf("Enter the numbers \n");
    for (i = 0; i < n; ++i)
        scanf("%d", &number[i]);

    for (i = 0; i < n; ++i)
    {

        for (j = i + 1; j < n; ++j)
        {

            if (number[i] > number[j])
            {

                a = number[i];
                number[i] = number[j];
                number[j] = a;
            }
        }
    }
}
```

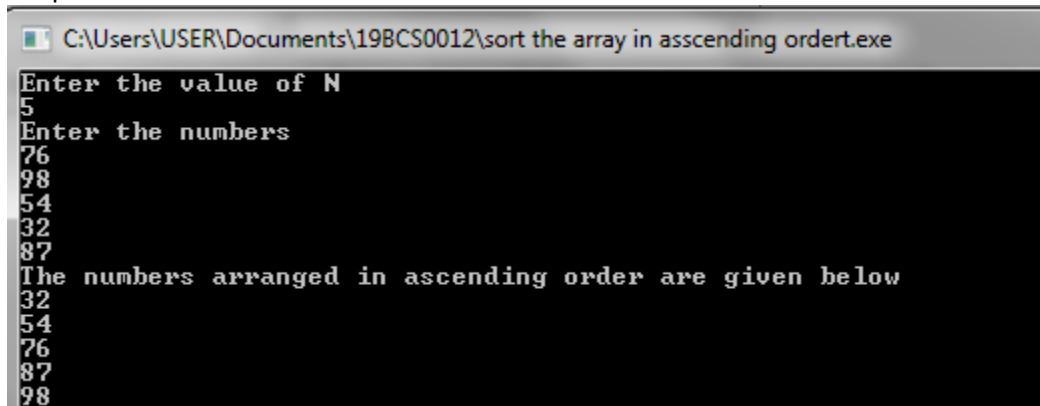
```

    }

}

printf("The numbers arranged in ascending order are given below \n");
for (i = 0; i < n; ++i)
    printf("%d\n", number[i]);
getch();
}
Output

```



The screenshot shows a Windows command prompt window with the title bar "C:\Users\USER\Documents\19BCS0012\sort the array in asscending ordert.exe". The prompt displays the following text:

```

Enter the value of N
5
Enter the numbers
76
98
54
32
87
The numbers arranged in ascending order are given below
32
54
76
87
98

```

## 5.Addition of two matrix

```

#include <stdio.h>

int main()
{
    int m, n, c, d, first[10][10], second[10][10], sum[10][10];

    printf("Enter the number of rows and columns of matrix\n");
    scanf("%d%d", &m, &n);
    printf("Enter the elements of first matrix\n");

    for (c = 0; c < m; c++)
        for (d = 0; d < n; d++)
            scanf("%d", &first[c][d]);

    printf("Enter the elements of second matrix\n");

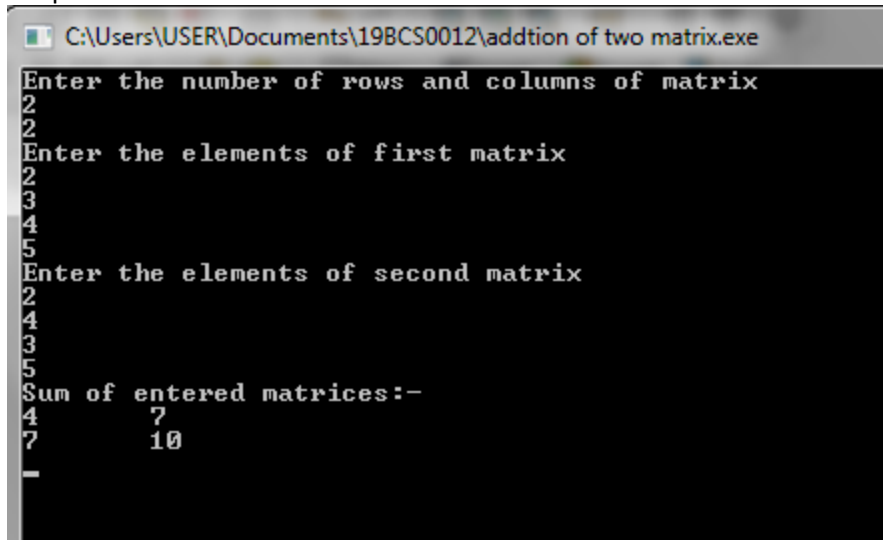
    for (c = 0; c < m; c++)
        for (d = 0; d < n; d++)
            scanf("%d", &second[c][d]);

    printf("Sum of entered matrices:-\n");
}

```

```
for (c = 0; c < m; c++) {  
    for (d = 0; d < n; d++) {  
        sum[c][d] = first[c][d] + second[c][d];  
        printf("%d\t", sum[c][d]);  
    }  
    printf("\n");  
}  
getch();  
return 0;  
}
```

Output



The screenshot shows a Windows command prompt window titled "C:\Users\USER\Documents\19BCS0012\addition of two matrix.exe". The program prompts the user to enter the number of rows and columns of the matrix. The user enters 2 for rows and 2 for columns. Then, the program prompts for the elements of the first matrix, and the user enters 2, 3, 4, and 5. Next, the program prompts for the elements of the second matrix, and the user enters 2, 4, 3, and 5. Finally, the program displays the sum of the entered matrices as follows:

```
Enter the number of rows and columns of matrix  
2  
2  
Enter the elements of first matrix  
2  
3  
4  
5  
Enter the elements of second matrix  
2  
4  
3  
5  
Sum of entered matrices:-  
4       7  
7       10  
-
```