**C PROGRAMMING ASSIGNMENT:**

**13**

DATE: 09.12.21

SUBMITTED BY: -

NAME: MUKTESH MISHRA

BRANCH: CSE

SECTION: B22

ROLL NO.: 21052258

1.WAP to find leader number by function definition.

***Code:***

#include <stdio.h>

*int* find\_max(*int* *n*)

{

*int* nextnum, largest = 0;

    for (*int* i = 0; i < *n*; i++)

    {

        printf("Enter next number\n");

        scanf("%d", &nextnum);

        if (nextnum > largest)

        {

            largest = nextnum;

        }

    }

    return (largest);

}

*int* countleader(*int* *n*)

{

*int* nextnum, largest = 0, c = 0;

    for (*int* i = 0; i < *n*; i++)

    {

        printf("Enter next number\n");

        scanf("%d", &nextnum);

        if (nextnum > largest)

        {

            largest = nextnum;

            c++;

        }

    }

    return (c);

    return (largest);

}

*int* main(*int* *argc*, *char* const \**argv*[])

{

*int* a;

    printf("Enter the value of n\n");

    scanf("%d", &a);

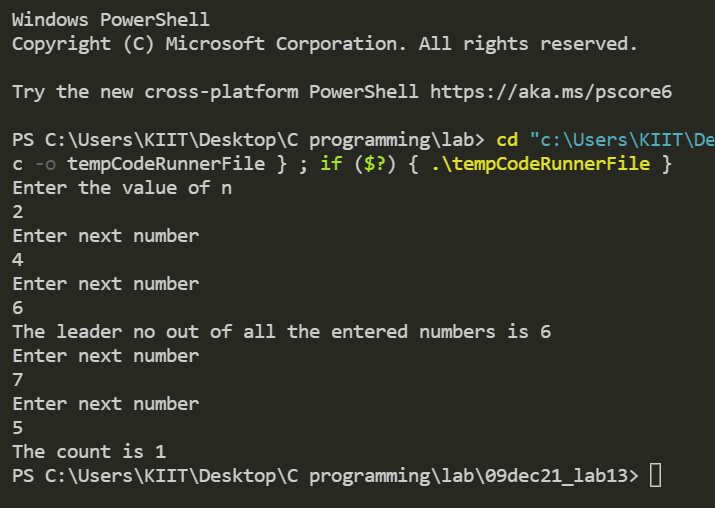
    printf("The leader no out of all the entered numbers is %d\n", find\_max(a));

    printf("The count is %d\n", countleader(a));

    return 0;

}

***Output:***



1. **WAP to find facorial using recursion**

***Code:***

#include <stdio.h>

*int* factorial(*int* *n*)

{

    if (*n* == 0)

        return 1;

    else

        return (*n* \* factorial(*n* - 1));

}

*int* main()

{

*int* number, fact;

    printf("Enter a number: ");

    scanf("%d", &number);

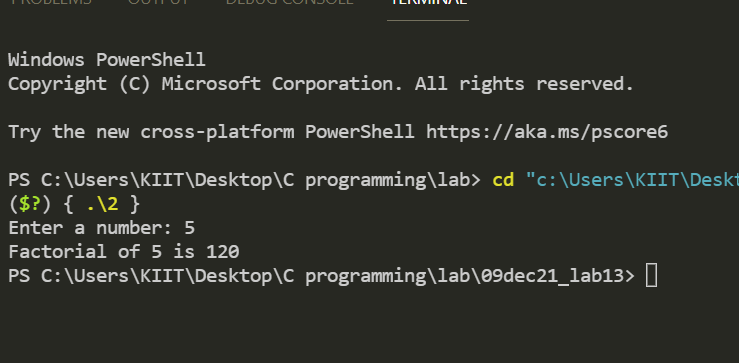
    fact = factorial(number);

    printf("Factorial of %d is %d\n", number, fact);

    return 0;

}

***Output:***



**3.WAP to find fibonacci series using recursion**

***Code:***

#include <stdio.h>

*int* Fibonacci(*int* *n*)

{

    if (*n* == 0)

        return 0;

    else if (*n* == 1)

        return 1;

    else

        return (Fibonacci(*n* - 1) + Fibonacci(*n* - 2));

}

*int* main()

{

*int* n, i = 0, c;

    printf("Enter n");

        scanf("%d", &n);

    printf("Fibonacci series\n");

    for (c = 1; c <= n; c++)

    {

        printf("%d\n", Fibonacci(i));

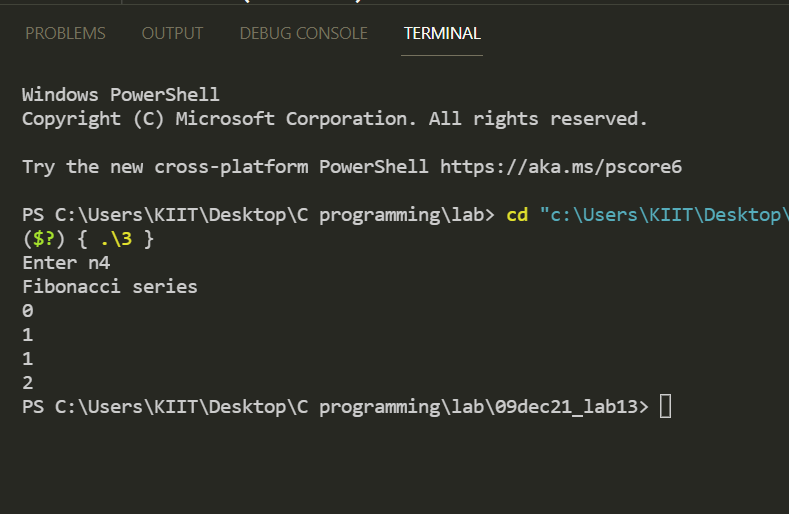
        i++;

    }

    return 0;

}

***Output***::



1. **WAP to find sum of digits using recursion**

***Code:***

#include <stdio.h>

*int* sum(*int* *num*)

{

    if (*num* != 0)

    {

        return (*num* % 10 + sum(*num* / 10));

    }

    else

    {

        return 0;

    }

}

*int* main()

{

*int* num, result;

    printf("Enter the number: ");

    scanf("%d", &num);

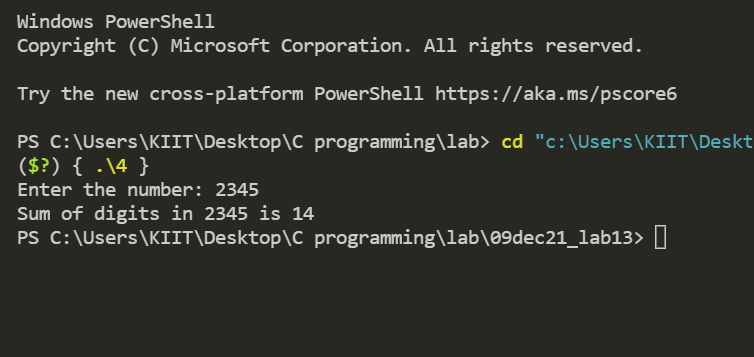
    result = sum(num);

    printf("Sum of digits in %d is %d\n", num, result);

    return 0;

}

***Output:***



5.WAP to find gcd using recursion

***Code:***

#include <stdio.h>

//gcd

*int* hcf(*int* *n1*, *int* *n2*)

{

    if (*n2* != 0)

        return hcf(*n2*, *n1* % *n2*);

    else

        return *n1*;

}

*int* main()

{

*int* n1, n2;

    printf("Enter two positive integers: ");

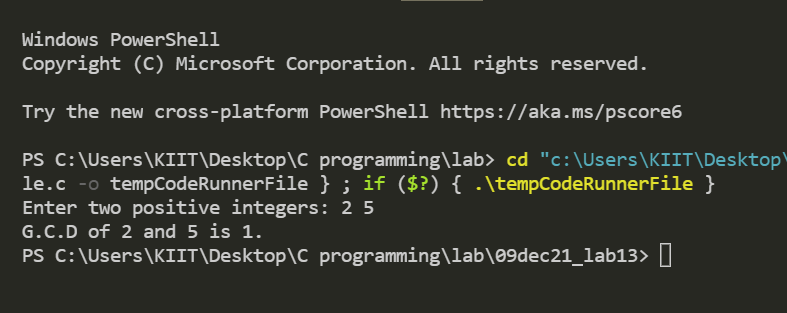
    scanf("%d %d", &n1, &n2);

    printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));

    return 0;

}

***Output:***



1. **WAP gcd of 3 nos by recursion**

***Code:***

#include <stdio.h>

//gcd of 3 nos by recursion

*int* gcd(*int* *n1*, *int* *n2*)

{

*int* t;

    if (*n1* % *n2* == 0)

        return *n2*;

    else

        return (t = gcd(*n2*, *n1* % *n2*));

}

*int* main()

{

*int* g, hcf, a, b, c;

    printf("Enter 3 numbers: ");

    scanf("%d %d %d", &a, &b, &c);

    g = gcd(a, b);

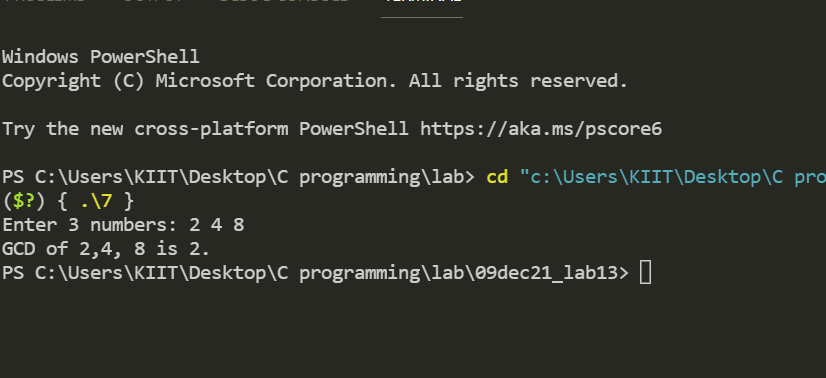
    hcf = gcd(g, c);

    printf("GCD of %d,%d, %d is %d.", a, b, c, hcf);

    return 0;

}

***Output:***



1. **WAP to find x^y**

***Code:***

#include <stdio.h>

//x^y

*int* power(*int* *base*, *int* *a*)

{

    if (*a* != 0)

        return (*base* \* power(*base*, *a* - 1));

    else

        return 1;

}

*int* main()

{

*int* base, a, result;

    printf("Enter base number: ");

    scanf("%d", &base);

    printf("Enter power number(positive integer): ");

    scanf("%d", &a);

    result = power(base, a);

    printf("%d^%d = %d", base, a, result);

    return 0;

}

***Output:***

