ACADEMY OF TECHNOLOGY

NAME : PRIYADARSHAN GHOSH

COLLEGE ROLL NO: 72

UNIVERSITY ROLL NO: 16900319072

DEPARTMENT: ECE-1(Y)

SEMESTER: 3rd

PAPER CODE : ES-CS391

➢ Laboratory *Assignment 2*

Q1. Write a C program to find power of any number using recursion.

Ans:

```
#include <stdio.h>
double pow(double base, int expo);
int main()
{
    double base, power;
    int expo;
    printf("Enter base: ");
    scanf("%lf", &base);
    printf("Enter exponent: ");
    scanf("%d", &expo);

    power = pow(base, expo);

    printf("%.2lf ^ %d = %f", base, expo, power);

    return 0;
}

double pow(double base, int expo)
{

    if(expo == 0)
        return 1;
    else if(expo > 0)
```

```
        return base * pow(base, expo - 1);
    else
        return 1 / pow(base, -expo);
}
```

OUTPUT =>

```
Enter base: 25
Enter exponent: 2
25.00 ^ 2 = 625.000000

...Program finished with exit code 0
Press ENTER to exit console.
```

Q2. WRITE A C PROGRAM TO PRINT ALL EVEN OR ODD NUMBERS IN GIVEN RANGE USING RECURSION.

Ans:

```
#include <stdio.h>
void printEvenOdd(int cur, int limit);

int main()
{
    int lowerLimit, upperLimit;
    printf("Enter lower limit: ");
    scanf("%d", &lowerLimit);
    printf("Enter upper limit: ");
    scanf("%d", &upperLimit);
```

```
    printf("Even/odd Numbers from %d to %d are: ", lowerLimit, upperLimit);
    printEvenOdd(lowerLimit, upperLimit);

    return 0;
}

void printEvenOdd(int cur, int limit)
{
    if(cur > limit)
        return;

    printf("%d, ", cur);

    printEvenOdd(cur + 2, limit);
}
```

OUTPUT =>

```
Enter lower limit: 5
Enter upper limit: 100
Even/odd Numbers from 5 to 100 are: 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 5
9, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99,

...Program finished with exit code 0
Press ENTER to exit console.
```

Q3. Write a C program to find sum of all-natural numbers between 1 to n using recursion.

Ans:

```
#include <stdio.h>
int sumOfNaturalNumbers(int start, int end);
int main()
{
    int start, end, sum;

    printf("Enter lower limit: ");
    scanf("%d", &start);
    printf("Enter upper limit: ");
    scanf("%d", &end);

    sum = sumOfNaturalNumbers(start, end);

    printf("Sum of natural numbers from %d to %d = %d", start, end,
sum);

    return 0;
}

int sumOfNaturalNumbers(int start, int end)
{
    if(start == end)
        return start;
```

```
    else
        return start + sumOfNaturalNumbers(start + 1, end);
}
```

OUTPUT =>

```
Enter lower limit: 2
Enter upper limit: 200
Sum of natural numbers from 2 to 200 = 20099

...Program finished with exit code 0
Press ENTER to exit console.
```

Q4. WRITE A C PROGRAM TO FIND SUM OF ALL EVEN OR ODD NUMBERS IN GIVEN RANGE USING RECURSION.

Ans:

```
#include <stdio.h>
int sumOfEvenOdd(int start, int end);
int main()
{
    int start, end, sum;
    printf("Enter lower limit: ");
    scanf("%d", &start);
    printf("Enter upper limit: ");
    scanf("%d", &end);
```

```
    printf("Sum of even/odd numbers between %d to %d = %d\n",
start, end, sumOfEvenOdd(start, end));


    return 0;
}
int sumOfEvenOdd(int start, int end)
{


    if(start > end)
        return 0;
    else
        return (start + sumOfEvenOdd(start + 2, end));
}
```

OUTPUT =>

```
Enter lower limit: 2
Enter upper limit: 220
Sum of even/odd numbers between 2 to 220 = 12210


...Program finished with exit code 0
Press ENTER to exit console.
```

Q5. WRITE A C PROGRAM TO FIND REVERSE OF ANY NUMBER USING RECURSION.

Ans:

```
#include<stdio.h>
void reverse_num(int num);
int main(void)
```

```
{
    int num;

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

    reverse_num(num);

    return 0;
}

void reverse_num(int num)
{
    int rem;
    if (num == 0)
    {
        return;
    }

    else
    {
        rem = num % 10;
        printf("%d", rem);
        reverse_num(num/10);
    }

}
```

OUTPUT =>

```
Enter a number: 987456

654789


...Program finished with exit code 0

Press ENTER to exit console.
```

Q6.WRITE A C PROGRAM TO CHECK WHETHER A NUMBER IS PALINDROME OR NOT USING RECURSION.

Ans:

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

int reverse(int num);
int isPalindrome(int num);

int main()
{
    int num;
    printf("Enter any number: ");
    scanf("%d", &num);

    if(isPalindrome(num) == 1)
    {
        printf("%d is palindrome number.\n", num);
    }
    else
```

```
    {
        printf("%d is NOT palindrome number.\n", num);
    }

    return 0;
}

int isPalindrome(int num)
{

    if(num == reverse(num))
    {
        return 1;
    }

    return 0;
}
int reverse(int num)
{
    int digit = (int)log10(num);


    if(num == 0)
        return 0;

    return ((num%10 * pow(10, digit)) + reverse(num/10));
}
```

OUTPUT =>

```
Enter any number: 48984
48984 is palindrome number.


...Program finished with exit code 0
Press ENTER to exit console.
```

Q7. WRITE A C PROGRAM TO FIND SUM OF DIGITS OF A GIVEN NUMBER USING RECURSION.

Ans:

```
#include <stdio.h>
int sum (int a);
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;
}

int sum (int num)
{
    if (num != 0)
    {
        return (num % 10 + sum (num / 10));
```

```
    }
    else
    {
       return 0;
    }
}
```

OUTPUT =>

```
 Enter the number: 78956
Sum of digits in 78956 is 35


...Program finished with exit code 0
Press ENTER to exit console.
```

Q8. WRITE A C PROGRAM TO FIND FACTORIAL OF ANY NUMBER USING RECURSION.

Ans:

```
#include<stdio.h>
long int multiplyNumbers(int n);
int main() {
    int n;
    printf("Enter a positive integer: ");
    scanf("%d",&n);
    printf("Factorial of %d = %ld", n, multiplyNumbers(n));
    return 0;
}
```

```
long int multiplyNumbers(int n) {
    if (n>=1)
        return n*multiplyNumbers(n-1);
    else
        return 1;
}
```

OUTPUT =>

```
Enter a positive integer: 11
Factorial of 11 = 39916800

...Program finished with exit code 0
Press ENTER to exit console.
```

Q9. WRITE A C PROGRAM TO GENERATE NTH FIBONACCI TERM USING RECURSION.

Ans:

```
#include <stdio.h>
int fibo(int);
int main()
{
    int num;
    int result;

    printf("Enter the nth number in fibonacci series: ");
    scanf("%d", &num);
    if (num < 0)
```

```
    {
        printf("Fibonacci of negative number is not possible.\n");
    }
    else
    {
        result = fibo(num);
        printf("The %d number in fibonacci series is %d\n", num,
result);
    }
    return 0;
}
int fibo(int num)
{
    if (num == 0)
    {
        return 0;
    }
    else if (num == 1)
    {
        return 1;
    }
    else
    {
        return(fibo(num - 1) + fibo(num - 2));
    }
}
```

OUTPUT =>

```
Enter the nth number in fibonacci series: 9
The 9 number in fibonacci series is 34

...Program finished with exit code 0
Press ENTER to exit console.
```

Q10. WRITE A C PROGRAM TO FIND GCD (HCF) OF TWO NUMBERS USING RECURSION.

Ans:

```
#include <stdio.h>
int gcd(int a, int b);
int main()
{
   int num1, num2, hcf;

   printf("Enter any two numbers to find GCD: ");
   scanf("%d%d", &num1, &num2);

   hcf = gcd(num1, num2);

   printf("GCD of %d and %d = %d", num1, num2, hcf);

   return 0;
}
int gcd(int a, int b)
```

```
{
    if(b == 0)
        return a;
    else
        return gcd(b, a%b);
}
```

OUTPUT =>

```
Enter any two numbers to find GCD: 6
8
GCD of 6 and 8 = 2


...Program finished with exit code 0
Press ENTER to exit console.
```