COURSE: DATA STRUCTURES AND ALGORITHMS

LECTURER: ALBIN SHEMA

RECURSION MID SEMESTER PREPARATIONS

1. Write a program in C to print first 50 natural numbers using recursion.
2. Write a program in C to calculate the sum of numbers from 1 to n using recursion.
3. Write a program in C to Print Fibonacci Series using recursion. Input number of terms for the Series
4. Write a program in C to print the array elements using recursion.
5. Write a program in C to count the digits of a given number using recursion.
6. Write a program in C to find the sum of digits of a number using recursion.
7. Write a program in C to find GCD of two numbers using recursion.
8. Write a program in C to get the largest element of an array using recursion.
9. Write a program in C to reverse a string using recursion.
10. Write a program in C to find the Factorial of a number using recursion.
11. Write a program in C to convert a decimal number to binary using recursion.
12. Write a program in C to check a number is a prime number or not using recursion.
13. Write a program in C to input two numbers and find the LCM of those two numbers using recursion.

14. Write a program in C to print even or odd numbers in given range using recursion.

15. Write a program in C to calculate the power of any number using recursion.

Predict the output of the following programs:

```
#include<stdio.h>
void k(int i,int j)
{
   static char x[]="0123456789ABCDEF";
   if(i>=j){
      k(i/j,j);
   }
   putchar(x[i%j]);
}
void main()
{
   k(20,12);
}
```

```
#include<stdio.h>
int test(int a,int b)
{
   if(a==b) return 1;
   else if(a>b)return 0;
   else return(a+test(a+1,b));
}
main()
{
   int x;
   x=test(0,3);
   printf("%d",x);
}
```

```
#include<stdio.h>
main()
{
   int x(int,int);
   printf("%d\n", x(1,1));
}
int x(int m,int n)
{
   if(m==0)return n+1;
   if(m>=1&&n==0)
      return x(m-1,1);
   else
      return x(m-1,x(m,n-1));
}
```

```
#include<stdio.h>
#define MAIN() main()
#define mAIN() Main()
int MaIn() {printf("Main ");}
int mAIN() {printf("main ");}
int MAIN() {printf("MAIN "); mAIN();}
```

```
#include<stdio.h>
void foo(int n)
{
   int m=1;
   if(n)
   {
      m++;
      foo(n+1);
   }
   m++;
   printf("m=%d, n=%d\n", m, n);
}

main()
{
   foo(-3);
}
```

```
#include<stdio.h>
#define max(a,b) a>b?a:b
main()
{
   int m,n;
   m=3+max(2,3);
   n=2*max(3,2);
   printf("%d %d", m, n);
}
```

```
#include<stdio.h>
int zap(int n)
{
   if(n<=1)
      return 1;
   else
      return(zap(n-3)+zap(n-1));
}
main()
{
   printf("%d\n",zap(6));
}
```

```
#include<stdio.h>
void fun(int x)
{
  if(x > 0)
  {
     fun(--x);
     printf("%d\t", x);
     fun(--x);
  }
}

int main()
{
  int a = 4;
  fun(a);
  return 0;
}
```

```
#include<stdio.h>
int fun(int a[],int n)
{
  int x;
  if(n == 1)
    return a[0];
  else
    x = fun(a, n-1);
  if(x > a[n-1])
    return x;
  else
    return a[n-1];
}

int main()
{
  int arr[] = {12, 10, 30, 50, 100};
  printf(" %d ", fun(arr, 5));
  return 0;
}
```

```
#include<stdio.h>
int fun(int i)
{
  if ( i%2 ) return (i++);
  else return fun(fun( i - 1 ));
}

int main()
{
  printf(" %d ", fun(200));
  return 0;
}
```