**Time:2 hours** **End Semester Exam-II: MTH409** **Full Marks 40**

1. Let $a[0], a[1], \cdots, a[n-1]$ are real numbers. The *recursive* C function
   *float sum(float *a,int n)*
   returns the sum $a[0] + a[1] + a[2] + \cdots\cdots + a[n-1]$. Complete the details of the above function. [3]

2. Study the following program and write down the output. [5]

```
#include <stdio.h>
void mystery(int);
int main( )
{int n=4;
mystery(n);
}
void mystery(int n)
{ if(n<=0)return;
if(n%2==0)mystery(n/2);
printf(“%d\n”,n*n);
mystery(2*n-5);
}
```

3. Define a structure that can describe a point in 2D. A circle in 2D can be specified by its centre (a point) and radius. Define a structure that can describe a circle in 2D. Write a C function with prototype that does the following. It accepts a circle and a point as arguments. It returns $1$ if the point lies inside the circle but returns $0$ otherwise. [2+2+4]

4. Create a binary search tree with the character strings *roy*, *tiwari*, *shukla*, *sanki*, *meraj*, *das*, *bera*, *rao*. Generate the output of the postorder and preorder traversal. [4+2+2]

5. Write down the equivalent postfix expression [4]

$$(a + b * (c - d/a)) * (c + d)/(a - b * c)$$

6. Write a program which does the following. It reads integer $n$ repeatedly until $n > 0$ is satisfied. It then calculates the number of integers between $1$ and $n$ that are divisible either by $3$ or by $8$ but not by both. [3]

7. Starting with initial guess $x_0$, the Newton method for finding the root of $f(x) = 0$ uses the iteration

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \qquad n = 0, 1, 2, \cdots,$$

where $f'$ denotes the derivative of $f$. Write a program using Newton method to calculate a root of the equation $x^2 - 2 = 0$. The program incorporates the following steps/functions. Use a function *fu* that accepts real number $x$ and returns the value of $f(x)$ at x. Use a function *dfu* that accepts real number $x$ and returns the value of $f'(x)$ at x. Read a real numbers $x_0$. Use a function *newton* which accepts real arguments $x_0$,*eps* and integer argument $ITMX$ and returns the root of the equations. Here *eps* is $10^{-5}$ and $ITMAX = 100$. The function *newton* uses the iteration described above until $|x_{n+1} - x_n| < eps$ or $n > ITMAX$. If $|x_{n+1} - x_n| < eps$ then it returns $x_n$ as root. If $n > ITMAX$, it prints an error message "Newton method does not converge in 100 iterations". [9]