Course: B. Sc. Hons (Mathematics)

Semester: IV (Fourth)

Paper Code: MATH CC408

Paper Name: Numerical Methods

Topic: Practical Programs

Prepared by: Dr. Seema Mishra

Email id: seema.mishra258@gmail.com

Practical No. 1

Objective: To calculate the sum

$$\frac{1}{1}+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{n}$$

for at least two different value of n.

Input:

```
n=100;
sum=0;
for i=1: n
   sum=sum+1/i;
end
fprintf ('The sum of the series is %f\n', sum)
```

Output:

For n=100

The sum of the series is 5.187378

For n=1000

The sum of the series is 7.485471

Practical No. 2

Objective:

(i) To find the absolute value of an integer.
(ii) To enter up to 100 integers in an array and sort it in ascending order.

Input:

(i)

```
n=input ('Enter an integer:');
abs_n=abs(n);
fprintf ('The absolute value of entered integer is %d\n', abs_n)
```

(ii)

```
a=[];
for i=1:100
        b=input('Enter a number or enter 100 to stop: ');
        if b==100
           break
        else
           a(i)=b;
        end
end
disp('Entered array is')
disp(a)
disp('Sorted array is')
c=sort(a);
disp(c)
```

Output:

(i)

```
Enter an integer: -67
The absolute value of the entered integer is 67
```

(ii)

```
Enter a number or enter 100 to stop: 56
Enter a number or enter 100 to stop: 87
Enter a number or enter 100 to stop: 89
Enter a number or enter 100 to stop: 32
Enter a number or enter 100 to stop: 100

Entered array is
56    87    89    32
 Sorted array is
32    56    87    89
```

Practical No. 3

Objective: To write a program for finding a root of the equation $x^3 + x - 1 = 0$ by using Bisection method.

Input:

```
syms x
f(x)=x^3+x-1
a=input('Enter the value of a: ')
b=input('Enter the value of b: ')
if f(a)*f(b)>0
   disp('Wrong Choice')
else
   p=(a+b)/2;
   err=abs(f(p));
   while err>1e-7
      if f(a)*f(p)<0
         b=p;
      else
         a=p;
      end
      p=(a+b)/2;
      err=abs(f(p));
   end
end
fprintf('A root of the equation is %f\n', p)
```

Output:

```
Enter the value of a: 0
Enter the value of b: 1
A root of the equation is 0.682328
```

Practical No. 4

Objective: To write a program for finding a root of the equation

$$xe^x = 1$$

using the Newton Raphson's method.

Input:

```
syms x
f(x)=x*exp(x)-1
g(x)=diff(f(x));
n=input('Enter the no. of decimal places: ');
epsilon=5*10^-(n+1)
x0=input('Enter initial approximation: ');
for i=1:100
    f0=f(x0);
    f0_der=g(x0);
    y=vpa(x0-f0/f0_der);
    err=abs(y-x0);
    if err<epsilon
        break
    end
    x0=y;
end
disp('The required root is')
y=vpa(round(10^n*y)/10^n);
disp(y)
fprintf('Number of iterations required is %d\n',i)
```

Output:

```
f(x)=x*exp(x)-1
Enter the no. of decimal places: 6
epsilon=5.0000e-07
Enter the initial approximation: 0.5
The required root is
0.567143
Number of iterations required is 4.
```

Practical No. 5

Objective: To find y(12.5) from the following data using the Lagrange's interpolation formula:

x	12	13	14	16
y	5	6	9	11

Input:

```
syms x
x=[12 13 14 16]
y=[5 6 9 11]
sum=0;
a=12.5;
for i=1:length(x)
    N=1;
    D=1;
    for j=1:length(x)
        if j~=i
            N=N*(a-x(j));
            D=D*(x(i)-x(j));
        end
    end
    sum=sum+N/D*y(i);
end
fprintf('The required value of y at x=%f is %f\n',a, sum)
```

Output:

```
x =

    12    13    14    16


y =

     5     6     9    11
```

The required value of y at x=12.500000 is 5.093750

Practical No. 6

Objective: To find an interpolating polynomial by using Lagrange's interpolation formula from the following data:

x	-2	1	3	7
y	5	7	11	34

Input:

```
syms x
x_v=[-2 1 3 7]
y_v=[5 7 11 34]
sum=0;
for i=1:length(x_v)
    N=1;
    D=1;
    for j=1:length(x_v)
        if j~=i
            N=N*(x-x_v(j));
            D=D*(x_v(i)-x_v(j));
        end
    end
    sum=sum+N/D*y_v(i);
end
sum=simplify(sum);
disp('The interpolating polynomial is')
disp(sum)
```

Output:

```
x_v =

    -2     1     3     7


y_v =

     5     7    11    34

The interpolating polynomial is
(43*x^3)/1080 + (101*x^2)/540 + (793*x)/1080 + 1087/180
```

Practical No. 7

Objective: To solve the following system of linear equations by using LU decomposition method:

$$3x + y + 6z = 0$$
$$-6x - 16z = 4$$
$$8y - 17z = 17$$

Input:

```
A=[3,1,6; -6,0,-16; 0,8,-17]
B=[0;4;17]
L=zeros(3,3);
U=zeros(3,3);
for i=1:3
   L(i,i)=1;
end
for j=1:3
   U(1,j)=A(1,j);
end
L(2,1)=A(2,1)/A(1,1);
L(3,1)=A(3,1)/A(1,1);
U(2,2)=A(2,2)-U(1,2)*L(2,1);
L(3,2)=(A(3,2)-L(3,1)*U(1,2))/U(2,2);
U(2,3)=A(2,3)-U(1,3)*L(2,1);
U(3,3)=A(3,3)-L(3,1)*U(1,3)-L(3,2)*U(2,3);
disp('L =')
disp(L)
disp('U =')
disp(U)
Y=inv(L)*B;
X=inv(U)*Y;
fprintf('solution is x=%f, y=%f, z=%f\n', X(1), X(2), X(3))
```

Output:

```
A =

    3    1    6
   -6    0  -16
    0    8  -17
```

```
B =
    0
    4
   17

L =
    1    0    0
   -2    1    0
    0    4    1

U =
    3    1    6
    0    2   -4
    0    0   -1

solution is x=2.000000, y=0.000000, z=-1.000000
```
