NAME-SHUBHOJYOTI ROY

ROLL NO-21CE8036

SUBJECT CODE-CES651(COMPUTATIONAL LABORATORY)

```
1(i) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
C=A+B;
disp(C)
```

```
(ii) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
D=A-B;
disp(D)
```

```
(iii) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
P=A*B;
disp(P)
```

```
(iv) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
Q=A*B;
disp(Q)
```

```
(v) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
C=trace(A);
disp(C)
```

```
(vi) A = [2 -3 1; 2 6 -8; 7 0 1];


B = [1 2 0; -1 1 2; 3 1 4];
C=prod(A,"all");
disp(C)
```

```
(vii) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
C=prod(diag(B));
disp(C)
```

```
(viii) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
C=prod(diag(B));
disp(C)
(ix) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
n=input("enter any no.");
C=B*n*B;
```

```
(x) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
C=B*transpose(A);
disp(C)
```

```
(xi) A = [2 -3 1; 2 6 -8; 7 0 1];
B = [1 2 0; -1 1 2; 3 1 4];
C=rank(B);
disp(C)
```

```
(xii) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
C=eig(A);
disp(C)
```

```
(xiii) A = [2 -3 1; 2 6 -8; 7 0 1];

B = [1 2 0; -1 1 2; 3 1 4];
C=inv(B);
disp(C)
```

2(i)

Write a program for the summation of 12 integer numbers.

```
sum=0;
 for  i=1:12
t=input('Enter number: ');
sum=sum+t;
 end
 fprintf('Sum: %d\n', sum)
```

OUTPUT:

```
Enter number: 1
Enter number: 2
Enter number: 3
Enter number: 4
Enter number: 5
Enter number: 6
Enter number: 7
Enter number: 8
Enter number: 9
Enter number: 10
Enter number: 11
Enter number: 12
SUM:78
```

(ii) Write a program for the summation of 1.5+3.2+4.1+7.9.

```
 sum_real_numbers = 1.5 + 3.2 + 4.1 + 7.9;
disp(['Sum of real numbers: ' num2str(sum_real_numbers)])
```

OUTPUT: 16.7

(iii) Write a program for multiplication of 7 real numbers.

```
prod=1;

 for i=1:7
 t=input('Enter number: ');
prod=prod*t;
 end
fprintf('Product: %d\n', prod);
```

OUTPUT: Enter number: 1

```
Enter number: 2
Enter number: 3
Enter number: 4
Enter number: 5
Enter number: 6
Enter number: 7
PROD=5040
```

(iv) Write a program for division of 2 integers.

```
 a=input("enter one number")

b=input("enter second number")
c=a/b;
disp(c);

OUTPUT:1.5
```

(v) Write a program to find out the remainder in the division of two numbers

```
a=input("enter one number");

b=input("enter second number");
c=mod(a,b);
disp(c)

OUTPUT:3
```

(vi) Write a program to find out (x-y)3.

```
a=input("enter any number")

b=input("enter other number")
c=(a+b)^3;
disp(c)
 OUTPUT:
enter any number 2
enter other number 3
C=125;
```

(vii) l= 3.1; m= -1.5; n= 3.14/2; x= 2/ 3.14; y= 3^(1/2);

```
a=(l^2)+m*x;
b=sin(n+(y/n));
c=1/(cos(n+log(x)));
d=(l+n)/(x+y);
```

```
e=(l+n)^3/m;
fprintf('(l^2)+mx :%d \n',a);
fprintf('sin(n+(y/n)):%d\n',b);
fprintf('1/(cos(n+ln(x))) :%d\n',c);
fprintf('(l+n)/(x+y):%d\n',d);
fprintf('(l+n)^3/m:%d\n',e);
```

```
OUTPUT:- (l^2)+mx :8.654586e+00
sin(n+(y/n)):4.514374e-01
1/(cos(n+ln(x))) :2.290162e+00
(l+n)/(x+y):1.971301e+00
(l+n)^3/m:-6.789838e+01
```

(viii) Write a program for the summation of first n integers: n is an input from user.

```
 n=input("enter one number");

sum=0;
for a=1:n
    sum=sum+a;
end
disp(sum);
```

(ix) Write a program for summation of such integers in between 1 and 100 which are divisible by 5 and not divisible by 8: Use Loop and IF THEN-ELSE-ENDIF statement.

```
 sum = 0;


for i = 1:100


    if mod(i, 5) == 0 && mod(i, 8) ~= 0


        sum = sum + i;
    end
end

disp(['The summation of numbers between 1 and 100 that are divisible by 5 but not divisible by 8 is: ' num2str(sum)]);
```

```
OUTPUT:- 930
```

(x) Write a program to calculate: 13+23+33+53+83+133+……. + up to nth term where n is input (Use dynamic memory allocation).

```
 sum= 0;

  n=  input('enter the value of n:');



    for i = 1:n
        sum  = sum + i^3;
    end
    disp(sum)
```

```
  OUTPUT:- enter the value of n:2
     9
  enter the value of n:4
   100
```

(xi) Write a program that calculates the difference in volume between two cones. Use subroutine to calculate the volume.

```
r1=input('Enter radius 1: ');
r2=input('Enter radius 2: ');
diff=abs(calcVolume(r1)-calcVolume(r2));
fprintf('Difference: %d\n', diff);
function v=calcVolume(r)
 v=4/3*pi*r^3;
end
```

```
OUTPUT:- Enter radius 1: 4

Enter radius 2: 5

Difference: 2.555162e+02

Enter radius 1: 6

Enter radius 2: 5

Difference: 3.811799e+02
```

3(i) Simply supported beam: Deflection at center and rotations at the ends.

```
 input('Enter the length of the beam (L): ');

UDL = input('Enter the uniformly distributed load (UDL): ');
E = input('Enter the modulus of elasticity (E): ');
I = input('Enter the moment of inertia (I): ');
```

```
deflection_center = (5 * UDL * L^4) / (384 * E * I);
rotation_end1 = (UDL * L^3) / (16 * E * I);
rotation_end2 = -rotation_end1;
disp(['Deflection at the center of the beam: ' num2str(deflection_center)]);
disp(['Rotation at end 1 of the beam: ' num2str(rotation_end1)]);
disp(['Rotation at end 2 of the beam: ' num2str(rotation_end2)]);
```

```
OUTPUT:- Enter the length of the beam (L): 10
Enter the uniformly distributed load (UDL): 20
Enter the modulus of elasticity (E): 1
Enter the moment of inertia (I): 1
Deflection at the center of the beam: 2604.1667
Rotation at end 1 of the beam: 1250
Rotation at end 2 of the beam: -1250

Enter the length of the beam (L): 20
Enter the uniformly distributed load (UDL): 40
Enter the modulus of elasticity (E): 2
Enter the moment of inertia (I): 2
Deflection at the center of the beam: 20833.3333
Rotation at end 1 of the beam: 5000
Rotation at end 2 of the beam: -5000
```

(ii) Cantilever beam: Deflection as well as rotation at the center and at the free end.

```
 L = input('Enter the length of the cantilever beam (L): ');

UDL = input('Enter the uniformly distributed load (UDL): ');
E = input('Enter the modulus of elasticity (E): ');
I = input('Enter the moment of inertia (I): ');
deflection_center = (UDL * L^4) / (8 * E * I);
rotation_center = (UDL * L^3) / (6 * E * I);
deflection_free_end = (UDL * L^4) / (24 * E * I);
rotation_free_end = (UDL * L^3) / (6 * E * I);
disp(['Deflection at the center of the cantilever beam: '
num2str(deflection_center)]);
disp(['Rotation at the center of the cantilever beam: ' num2str(rotation_center)]);
disp(['Deflection at the free end of the cantilever beam: '
num2str(deflection_free_end)]);
disp(['Rotation at the free end of the cantilever beam: '
num2str(rotation_free_end)]);
```

```
OUTPUT:-
Enter the length of the cantilever beam (L): 10
Enter the uniformly distributed load (UDL): 20

Enter the modulus of elasticity (E): 2

Enter the moment of inertia (I): 2

Deflection at the center of the cantilever beam: 6250

Rotation at the center of the cantilever beam: 833.3333

Deflection at the free end of the cantilever beam: 2083.3333

Rotation at the free end of the cantilever beam: 833.3333
```

```
Enter the length of the cantilever beam (L): 20

Enter the uniformly distributed load (UDL): 10

Enter the modulus of elasticity (E): 1

Enter the moment of inertia (I): 1

Deflection at the center of the cantilever beam: 200000

Rotation at the center of the cantilever beam: 13333.3333

Deflection at the free end of the cantilever beam: 66666.6667

Rotation at the free end of the cantilever beam: 13333.3333
```

(iii) Fixed beam: Deflection as well as rotation at 0.25L and 0.75L from the left end.

```
 L = input('Enter the length of the fixed beam (L): ');

UDL = input('Enter the uniformly distributed load (UDL): ');
E = input('Enter the modulus of elasticity (E): ');
I = input('Enter the moment of inertia (I): ');
deflection_mid = (5 * UDL * L^4) / (384 * E * I);
rotation_025L = (UDL * L^3) / (24 * E * I);
rotation_075L = rotation_025L;
disp(['Deflection at the mid-span of the fixed beam: ' num2str(deflection_mid)]);
disp(['Rotation at 0.25L from the left end: ' num2str(rotation_025L)]);
disp(['Rotation at 0.75L from the left end: ' num2str(rotation_075L)]);
```

```
OUTPUT:- Enter the length of the fixed beam (L): 10
Enter the uniformly distributed load (UDL): 10
Enter the modulus of elasticity (E): 1
Enter the moment of inertia (I): 1
Deflection at the mid-span of the fixed beam: 1302.0833
```

```
Rotation at 0.25L from the left end: 416.6667
Rotation at 0.75L from the left end: 416.6667

Enter the length of the fixed beam (L): 20
Enter the uniformly distributed load (UDL): 20
Enter the modulus of elasticity (E): 2
Enter the moment of inertia (I): 2
Deflection at the mid-span of the fixed beam: 10416.6667
Rotation at 0.25L from the left end: 1666.6667
Rotation at 0.75L from the left end: 1666.6667
```

(iv) Deflection and rotation at any location of a determinate beam. Geometry, material property, and boundary condition as well as the location(s) will be user input.

```
L = input('Enter the length of the beam (L): ');
UDL = input('Enter the uniformly distributed load (UDL): ');
E = input('Enter the modulus of elasticity (E): ');
I = input('Enter the moment of inertia (I): ');
location = input('Enter the location along the beam (0 <= x <= L): ');
if location < 0 || location > L
    error('Invalid location. Please enter a value within the beam length.');
end
deflection_location = (UDL * location^2 * (3*L - location)) / (6 * E * I);
rotation_location = (UDL * location * (L - location)^2) / (2 * E * I);
disp(['Deflection at location ' num2str(location) ': '
num2str(deflection_location)]);
disp(['Rotation at location ' num2str(location) ': ' num2str(rotation_location)]);
```

```
OUTPUT:- Enter the length of the beam (L): 10

Enter the uniformly distributed load (UDL): 10

Enter the modulus of elasticity (E): 1

Enter the moment of inertia (I): 1

Enter the location along the beam (0 <= x <= L): 5

Deflection at location 5: 1041.6667

Rotation at location 5: 625


Enter the length of the beam (L): 20

Enter the uniformly distributed load (UDL): 20

Enter the modulus of elasticity (E): 1

Enter the moment of inertia (I): 1
Enter the location along the beam (0 <= x <= L): 30
```