NUMERICAL ANALYSIS LAB (GR20D5021)

Name of the Student: …………………………………………………………….

Branch: ………………………………………………………………………

Roll No: .……………………… …………… …**Academic Year:** ………………

GOKARAJU RANGARAJU

INSTITUTE OF ENGINEERING AND TECHNOLOGY

Nizampet Road, Bachupally, Hyderabad - 500 090

GOKARAJU RANGARAJU
INSTITUTE OF ENGINEERING AND TECHNOLOGY

Nizampet Road, Bachupally, Hyderabad - 500 090

CERTIFICATE

This is to certify that it is a bonafide record of practical work done in the ………………………………………………………………………………laboratory in ……………………… semester of ……………………… year during the academic year ………………by

Name of the Student : ………………………………………..

Roll. No. : ………………………………………..

Branch : ……………………………………….

Section : ………………………………...............

Signature of the Internal Examiner	**Signature of the Head of Department**	**Signature of the External Examiner**

Index

Exp. No.	Name of the Experiment	Date of Submission	Remarks	Signature of Lab In charge with date
1	Roots of Non-Linear Equation Using Bisection Method			
2	Roots of Non-Linear Equation Using Newton’s Method			
3	Curve Fitting by Least Square Approximations			
4	Solve the System of Linear Equations Using Gauss - Elimination Method			
5	Solve the System of Linear Equations Using Gauss - Seidal Iteration Method			
6	Solve the System of Linear Equations Using Gauss - Jorden Method			
7	Integrate numerically using Trapezoidal Rule			
8	Integrate numerically using Simpson’s Rule			
9	Numerical Solution of Ordinary Differential Equations by Euler’s Method			
10	Numerical Solution of Ordinary Differential Equations by Runge- Kutta Method			

Date:

EXPERIMENT – 01

TO FIND THE POINTS OF CONTRAFLEXURE FOR A BEAM USING BISECTION METHOD

Aim

To find the points of contraflexure for a beam using Bisection method

Tools required

Desktop, Dev C/C++

Theory

Bisection method is based on the repeated application of the intermediate value property. Let the function f(x) be continuous between a and b. For definiteness, let f(a) be negative and f(b) be positive. Then the first approximation to the root is $x_1 = (a+b)*0.5$.

If $f(x_1) = 0$ then x_1 is the root of f(x)=0. Otherwise, the root lies between a and x_1 or x_1 and b according as $f(x_1)$ is positive or negative. Then we bisect the interval as before and continue the process until the root is found to the desired accuracy.
If $f(x_1)$ is positive so that the root lies between a and x_1. Then the second approximation to the root is $x_1 = 0.5*(a+x_1)$. If $f(x_2)$ is negative then the root lies between x_1 and x_2. The third approximation to the root is $x_3 = 0.5*(x_1+x_2)$ and so on.
The error reduces by a factor of ½ each step, the error is linear.

Problem statement

A fixed beam AB of length 10 m carries a udl of 10kN/m over its entire span. Determine the end moments. Draw Shear Force Diagram and Bending Moment Diagram. Also, write a program in C/C++ to find the points of contraflexure using Bisection method.

Analytical Formulation

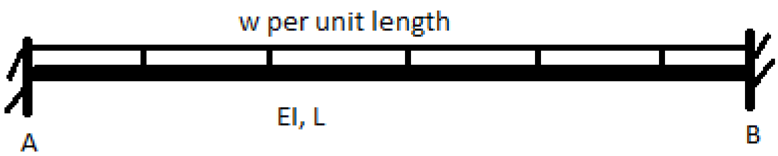

Fig.1.1 A fixed beam carrying udl over its entire length

FEM	$-\frac{wl^2}{12}$	$\frac{wl^2}{12}$
Reaction	$\frac{wl}{2}$	$\frac{wl}{2}$

Consider a section XX at a distance of x from left end A.

$$EI\,\frac{d^2y}{dx^2} = V_A x - \frac{wl^2}{12} - \frac{wx^2}{2} = \frac{wl}{2}x - \frac{wl^2}{12} - \frac{wx^2}{2}$$

To find the points of contraflexure, set $BM_{XX} = 0$

Find the value of x

a) Analytically

b) Verify your result using bisection method manually, the LL for x = 0, the UL for x is L/2.

Analytical result

Substituting the values, we have

$$f(x) = \frac{100x}{2} - \frac{1000}{12} - \frac{10x^2}{2}$$

To find the value of x, set f(x) = 0

$$x = \frac{-50 \pm \sqrt{50^2 - 4(-5)(-\frac{1000}{12})}}{2(-5)} = \frac{-50 \pm 28.867513}{-10} = 2.113248, 7.8867$$

The distance to the point of contraflexure from left end A is 2.113248 m and 7.8867 m

Bisection Method Manually

Perform at least Five iterations.

Code in ‘C’ language

```
/* preprocessing */
#include <stdio.h>
#include <math.h>

/* user defined */
float f(float x)
{
return (FUNCTION OF X);               // fixed beam
}

void bisect (float *x, float a, float b, int *itr)
{
*x = (a+b)/2;
++(*itr);
printf("Iteration no. %3d X = %7.5f\n", *itr, *x);
}

/* main */
main()
{
int itr=0; int maxitr;
float x, a, b, aerr, x1;
printf("Enter the values of a, b, allowed error, maximum iterations\n");
scanf("%f %f %f %d",&a,&b,&aerr,&maxitr);
bisect(&x,a,b,&itr);
/* 3 2 0.0001 20 */

do

{

if (f(a)*f(x) <0)
b =x  ;
else
a = x;
bisect(&x1,a,b,&itr);

if(fabs(x1-x) < aerr)
{
printf("After %d iterations, root is %12.7f\n",itr,x1);
return 0;
}
x = x1;
} while (itr < maxitr);

printf("Solution does not converge, iterations not sufficient");

return 1;

}
```

Result from the 'C' program

OUTPUT

RESULT

Using the bisection method, the distance to the point of contraflexure from the left end is _______ m

Problem 2
To find the roots of the equation using Bisection method

Aim: To locate the point(s) where slope is equal to zero for a simply supported beam AB of span 10 m carrying a concentrated load of 100 kN acting vertically downwards at a distance of 7m from the left end support A. Use bisection method.

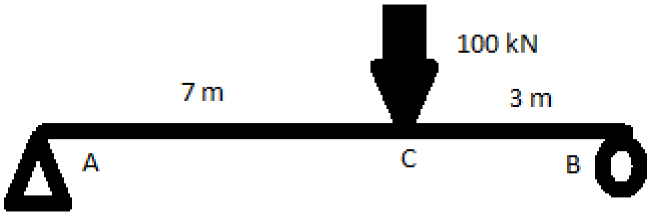

Fig1.2 A simply supported beam carrying a point load at C

Use second order method,

$$EI\,\frac{d^2y}{dx^2} = V_A x - W(x-a)$$

Integrating, we have

$$EI\,\frac{dy}{dx} = V_A\frac{x^2}{2} + C_1 - W\frac{(x-a)^2}{2}$$

$$EI\,y = V_A\frac{x^3}{6} + C_1 x + C_2 - W\frac{(x-a)^3}{6}$$

To find the constants of integration, we use the boundary conditions
At x= 0 y=0 we get C_2=0
At x = l0m y = 0

$$0 = 30 * \frac{10^3}{6} - 100\left(\frac{3^3}{6}\right) + 10C_1$$

$$C_1 = -455$$

$$y = f(x) = 30\frac{x^3}{6EI} - \frac{455x}{EI} - 100\frac{(x-7)^3}{6EI}$$

To find the point of zero slope of the curve, the function for slope is equal to

$$S(x) = 30\frac{x^2}{2EI} - \frac{455}{EI} - 100\frac{(x-7)^2}{2EI}$$

Step 1
Assume that the point of zero slope is in the span AC

$$0 \le x \le 7$$

Neglect the negative term in the bracket, we have

$$S(x) = 30\frac{x^2}{2EI} - \frac{455}{EI} = 0$$

Result

Analytical result is x = ___________ m

Use bisection method to find the root of the equation,
Starting value is x_0 =0 and x_1 = 7

C++ OUTPUT

RESULT
Using the bisection method, point(s) where slope is equal to zero **is** at a distance of ________m from left end.

Date:

EXPERIMENT – 02

TO FIND THE POINT OF ZERO SHEAR FORCE FOR A BEAM USING NEWTONS METHOD

Aim

To find the point of zero shear force for a beam using Newton's method.

Tools required

Desktop, Dev C/C++

Theory

Let x_0 be an approximate root of the equation f(x)=0. If $x_1 = x_0 + h$ be the exact root then $f(x_1)=0$.
Expanding by Taylor's series
Since h is small, neglecting the h^2 and higher powers of x, we have

$$f(x_0+h) = f(x_0) + hf'(x_0) + \frac{h^2}{2}f''(x_0) + \cdots = 0$$

$$f(x_0) + hf'(x_0) = 0$$

$$h = -\frac{f(x_0)}{f'(x_0)}$$

A closer approximation to the root is given by

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

repeat the iterations until convergence, until two successive values of x are same up to four decimals

Problem 1

To find the points of zero shear force using Newtons method for a fixed beam carrying a uvl as shown in the Fig.2.1 below.

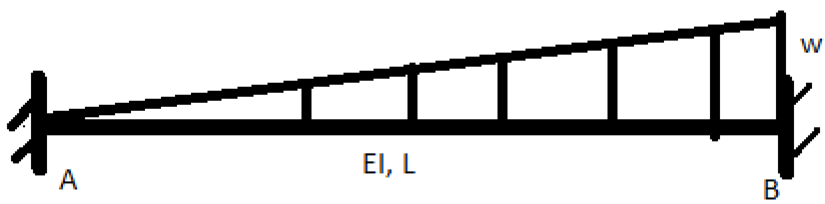

Fig. 2.1 A fixed beam carrying a uniformly varying load

End moments	$wl^2/30$	$wl^2/20$
Reactions	3wl/20	7wl/20

Using Macaulay's method, we have

$$BM(x) = EI\frac{d^2y}{dx^2} = \frac{3wl}{20}x - \frac{wl^2}{30}x^0 - \frac{1}{2}x.\frac{wx}{l}.\frac{x}{3}$$

Integrating, we have

$$SE(x) = EI\frac{dy}{dx} = \frac{\frac{3wl}{20}x^2}{2} - \frac{wl^2}{30}x^1 - \frac{1}{2}x.\frac{wx}{l}.\frac{x}{3}.\frac{x}{4}$$

$$SF(x) = EI\ \frac{d^3y}{dx^3} = \frac{3wl}{20} - \frac{1}{2}.\frac{wx}{l}.x$$

To find the point of zero shear force, equate SF(x)=0 and find the roots of the equation SF(x) using Newtons method.

To find the point of zero bending moment equate BM(x) = 0 and find the roots of the equation using Bisection method and Newtons method.

To find the point of zero slope equate SE(x) = 0 and find the roots of the equation using both Bisection method and Newtons method.

Problem 2
To find the roots of the equation using Bisection method and Newton's method

Aim: To locate the point(s) where slope is equal to zero for a simply supported beam AB of span 10 m carrying a udl of magnitude 10kN/m over a span of 7m from end A and a concentrated load of 100 kN acting vertically downwards at a distance of 7m from the left end support A. Use Newtons method.

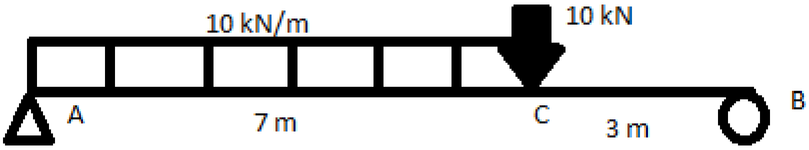

Fig. 3.2 A simply supported beam ACB

Result

Analytical result is x =

Use bisection method to find the root of the equation S(x)=0
Starting value is x_0 =3 and x_1 = 10

Write the **program in 'C'** and verify your result
A = 3.001
B = 10.000
Allowable error = 1e-5
Maximum iterations is 10000
Error calculation compare the analytical with the bisection/Newton method
Calculate the relative error and percentage error

Using Bisection Method

Using Newton's Method

Code in C for Newton's Method

```
/* Newton method */
/* preprocessing */
#include <stdio.h>
#include <math.h>

/*user defined functions*/
float f(float x)
{
return (-10*x*x*x/6 + 34.25*x*x - 15*x - 651.125);
//return 15*x*x-455;

//x*log10(x)-1.2;
}
float df(float x)
{
//   if(x<=7) {
//    return 30*x;
return (-5*x*x+68.5*x-15);
       //}
//   if(x>=7) { return 30*x-100*(x-7); }

//log10(x) + 0.43429;
}

/* main */
main()
{
int itr, maxitr;
 float h, x0, x1, aerr;
printf("Enter x0, allowed error, maximum iterations\n"); /* 0.01 0.000001 1000*/
scanf("%f %f %d",&x0, &aerr,&maxitr);
for(itr=1;itr<=maxitr;itr++)
{
```

```
h = f(x0)/df(x0);
x1 = x0-h;
printf("Iterations no. %3d,"
 "x = %9.6f\n",itr,x1);
if(fabs(h) < aerr)
{
printf("After %3d iterations,"
"root = %8.6f\n",itr, x1);
return 0;
}
x0=x1;
}
printf("Iterations not sufficient,"
"solution does not converge\n");
return 1;
}
```

OUTPUT

Date:

EXPERIMENT – 03

TO FIND THE DEFLECTION AT A SECTION OF A BEAM USING LEAST SQUARE APPROXIMATION

Aim

To find the deflection at a section of beam using least square approximation

Tools required

Desktop, Dev C/C++

Principle

By French Mathematician Adrien Marie Legendre in 1806

"The curve of best fit is that for which e's are as small as possible i.e., the sum of the squares of the errors is a minimum"

Example - Microsoft® Excel uses MLS to fit the trend line of degree 2.

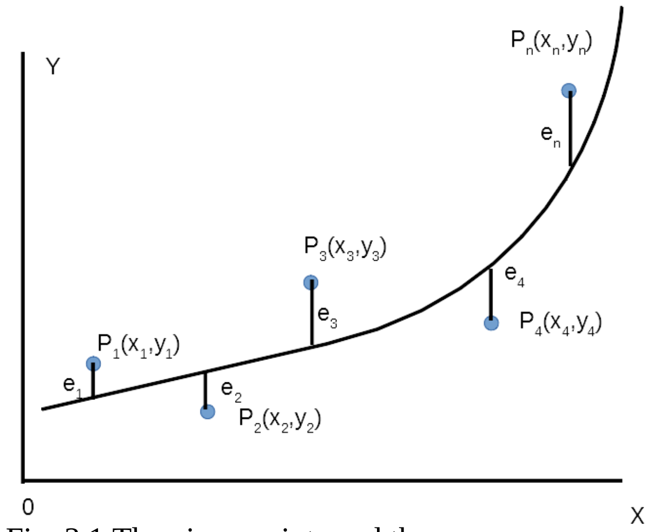

Fig. 3.1 The given points and the curve

Theory

The graphical methods has a drawback of giving a unique curve. The method of least squares provides an elegant procedure of fitting a unique curve to a given data.

Let the equation of the curve be $y = a + bx + cx^2$

$$E = e_1^2 + e_2^2 + e_3^2 + e_4^2 + \cdots + e_n^2$$

$$E = [y_1 - (a + bx_1 + cx_1{}^2)]^2 + [y_2 - (a + bx_2 + cx_2{}^2)]^2 + \cdots + [y_n - (a + bx_n + cx_n{}^2)]^2$$

The first derivative is set equal to zero

$$\frac{\partial E}{\partial a} = -2[y_1 - (a + bx_1 + cx_1{}^2)] \quad - 2[y_2 - (a + bx_2 + cx_2{}^2)] \quad - \cdots - 2[y_n - (a + bx_n + cx_n{}^2)]$$

$$\frac{\partial E}{\partial b} = -2x_1[y_1 - (a + bx_1 + cx_1{}^2)] \quad - 2x_2[y_2 - (a + bx_2 + cx_2{}^2)] \quad - \cdots - 2x_n[y_n - (a + bx_n + cx_n{}^2)]$$

$$\frac{\partial E}{\partial c} = -2x_1^2[y_1 - (a + bx_1 + cx_1{}^2)] \quad - 2x_2^2[y_2 - (a + bx_2 + cx_2{}^2)] \quad - \cdots - 2x_n^2[y_n - (a + bx_n + cx_n{}^2)]$$

$$\sum_i y_i = na + b\sum_i x_i + c\sum_i x_i{}^2$$

$$\sum_i x_i y_i = a\sum_i x_i + b\sum_i x_i^2 + c\sum_i x_i{}^3$$

$$\sum_i x_i^2 y_i = a\sum_i x_i^2 + b\sum_i x_i^3 + c\sum_i x_i{}^4$$

For the error E to be minimum, the second derivatives should be positive $\frac{\partial^2 E}{\partial a^2} > 0,\ \frac{\partial^2 E}{\partial b^2} > 0,\ \frac{\partial^2 E}{\partial c^2} > 0$

Problem statement

Aim: To perform curve fitting and determine the deflected shape of the curve. Find the deflection at a desired section on the deflected curve.

AB is simply supported beam having a length of 10m and flexural rigidity 'EI' as constant. The beam carries a uniformly distributed load of 10kN/m from the support A over a distance of 5m. To perform curve fitting using method of least squares and determine the equation of the deflected shape of the beam and find the deflection at any desired point on the curve.

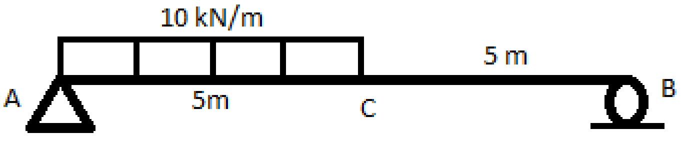

Fig.3.2 A simply supported beam ACB carrying a udl over span AC

The support reactions are V_B = 12.5 kN and V_A = 37.5 kN

Use the second order method, we have

Consider a section in the span AC at a distance of x from right end B

The bending moment at the section XX is given by

$$EI\frac{d^2y}{dx^2} = 12.5x - \frac{10(x-5)^2}{2}$$

$$EI\frac{dy}{dx} = 12.5\frac{x^2}{2} - \frac{10(x-5)^3}{6} + C_1$$

$$EI\ y = 12.5\ \frac{x^3}{6} - \frac{10(x-5)^4}{24} + C_1x + C_2$$

To find the constants of integration, we apply the boundary conditions

At x = 0 y = 0 $\quad$ C_2 = 0 $\quad$ At x = 10 y = 0 $\quad$ C_1 = -182.291667

X	Y	X	Y

0	0	5.5	-656.0155883
0.5	-90.88541333	6.0	-644.1666267
1.0	-180.2083267	6.5	-614.8697483
1.5	-266.40624	7.0	-568.1249533
2.0	-347.9166533	7.5	-504.5572417
2.5	-423.1770667	8.0	-425.4166133
3.0	-490.62498	8.5	-332.5780683
3.5	-548.6978933	9.0	-228.5416067
4.0	-595.8333067	9.5	-116.4322283
4.5	-630.46872	10.0	6.66667E-05
5.0	-651.0416333		

Result

Result using Macaulay's method

Analytical result using method of least squares – fit a parabola

Write a program in ‘C’ language

Code in ‘C’

```
/* parabolic fit by method of least squares */
#include <stdio.h>
main()
{
float augm[3][4] = {{0,0,0,0},{0,0,0,0},{0,0,0,0}};
float t,a,b,c,x,y,xsq;
int i,j,k,n;
puts("Enter the number of pairs of observed values:");
scanf("%d",&n);
augm[0][0]=n;
for(i=0;i<n;i++)
{
printf("pari no.%d\n",i+1);
scanf("%f %f",&x,&y);
xsq = x*x;
augm[0][1] += x;
augm[0][2] += xsq;
augm[1][2] += x*xsq;
augm[2][2] += xsq*xsq;
augm[0][3] += y;
augm[1][3] += x*y;
augm[2][3] += xsq*y;
}
augm[1][1] = augm[0][2];
augm[2][1] = augm[1][2];
augm[1][0] = augm[0][1];
augm[2][0] = augm[1][1];

puts("The augmented matrix is:-");
for(i=0;i<3;i++)
{
        for(j=0;j<4;j++)
        printf("%9.4f",augm[i][j]);
        printf("\n");
}

/* Now solving for a,b,c by Gauss Jordan Method */
for(j=0; j<3;j++)
        for(i=0;i<3;i++)
        if(i!=j)
        {
t = augm[i][j]/augm[j][j];
for(k=0;k<4;k++)
augm[i][k]-=augm[j][k]*t;
}
a = augm[0][3]/augm[0][0];
b = augm[1][3]/augm[1][1];
c = augm[2][3]/augm[2][2];
printf("a = %8.4f b= %8.4f  c = %8.4f\n",a,b,c);
}
```

Table 3.1 showing the data for the deflected curve using method of least squares

X	Y	SXY	SX2Y	SX2	SX3	SX4
0	0	0	0	0	0	0
0.5	-90.8854133	-45.4427067	-22.7213533	0.25	0.125	0.0625
1	-180.208327	-180.208327	-180.208327	1	1	1
1.5	-266.40624	-399.60936	-599.41404	2.25	3.375	5.0625
2	-347.916653	-695.833307	-1391.66661	4	8	16
2.5	-423.177067	-1057.94267	-2644.85667	6.25	15.625	39.0625
3	-490.62498	-1471.87494	-4415.62482	9	27	81
3.5	-548.697893	-1920.44263	-6721.54919	12.25	42.875	150.0625
4	-595.833307	-2383.33323	-9533.33291	16	64	256
4.5	-630.46872	-2837.10924	-12766.9916	20.25	91.125	410.0625
5	-651.041633	-3255.20817	-16276.0408	25	125	625
5.5	-656.015588	-3608.08574	-19844.4715	30.25	166.375	915.0625
6	-644.166627	-3864.99976	-23189.9986	36	216	1296
6.5	-614.869748	-3996.65336	-25978.2469	42.25	274.625	1785.0625
7	-568.124953	-3976.87467	-27838.1227	49	343	2401
7.5	-504.557242	-3784.17931	-28381.3448	56.25	421.875	3164.0625
8	-425.416613	-3403.33291	-27226.6633	64	512	4096
8.5	-332.578068	-2826.91358	-24028.7654	72.25	614.125	5220.0625
9	-228.541607	-2056.87446	-18511.8701	81	729	6561
9.5	-116.432228	-1106.10617	-10508.0086	90.25	857.375	8145.0625
10	6.67E-05	0.00066667	0.00666667	100	1000	10000
Total						
105	-8315.96284	-42871.0239	-260059.892	717.5	5512.5	45166.625

Substituting in the equations for the method of least squares, we have the simultaneous equations as

$$-8315.9628 = 21a + b(105) + c(717.5)$$
$$-42871.02386 = a(105) + b(717.5) + c(5512.5)$$
$$-260059.8916 = a(717.5) + b(5512.5) + c(45166.625)$$

Solving the simultaneous equations, for a, b, c we get the parabolic equation for the deflected curve of the beam is given by

$$55.167425 - 270.47246x + 26.37648857x^2$$

Table 3.2 Data using the equation from method of least squares

X	Y
0	55.167425
0.5	-73.474683
1	-188.928547
1.5	-291.194167
2	-380.271543
2.5	-456.160675
3	-518.861563
3.5	-568.374207
4	-604.698607
4.5	-627.834763
5	-637.782675
5.5	-634.542343
6	-618.113767
6.5	-588.496947
7	-545.691883
7.5	-489.698575
8	-420.517023
8.5	-338.147227
9	-242.589187
9.5	-133.842903
10	-11.908375

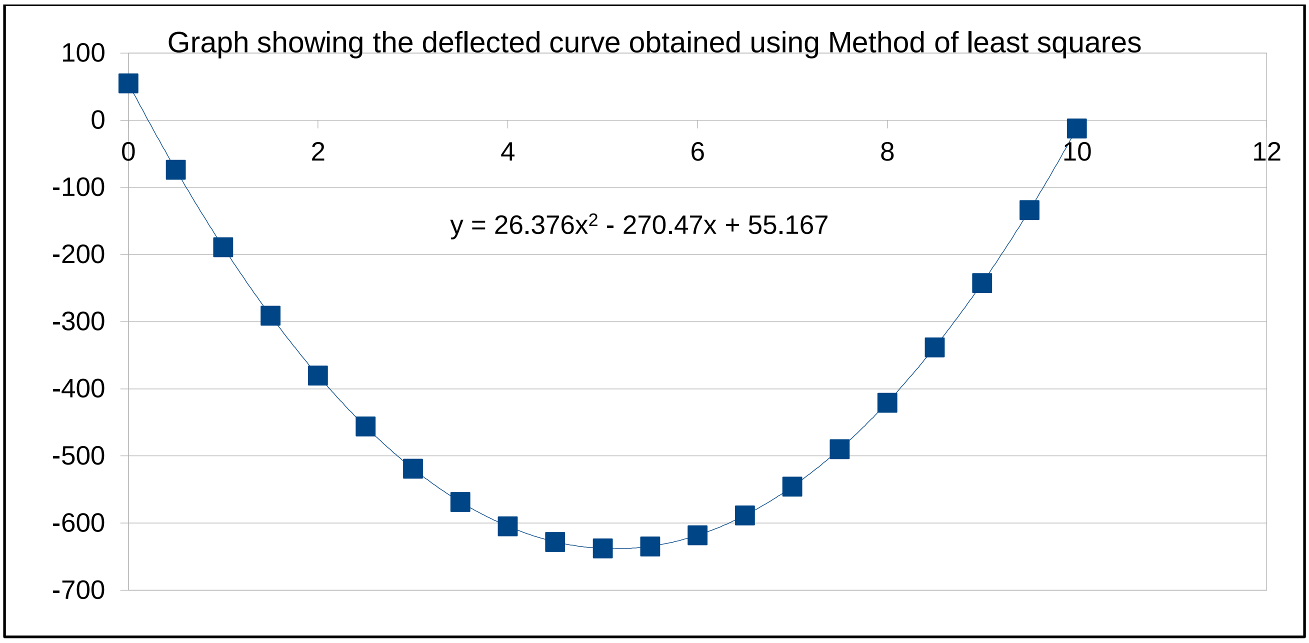

Fig.3.3 showing the deflected curve using method of least squares

Graph in MS Excel®

MS Excel® uses method of least squares to fit a parabola for a given set of (x, y) data points

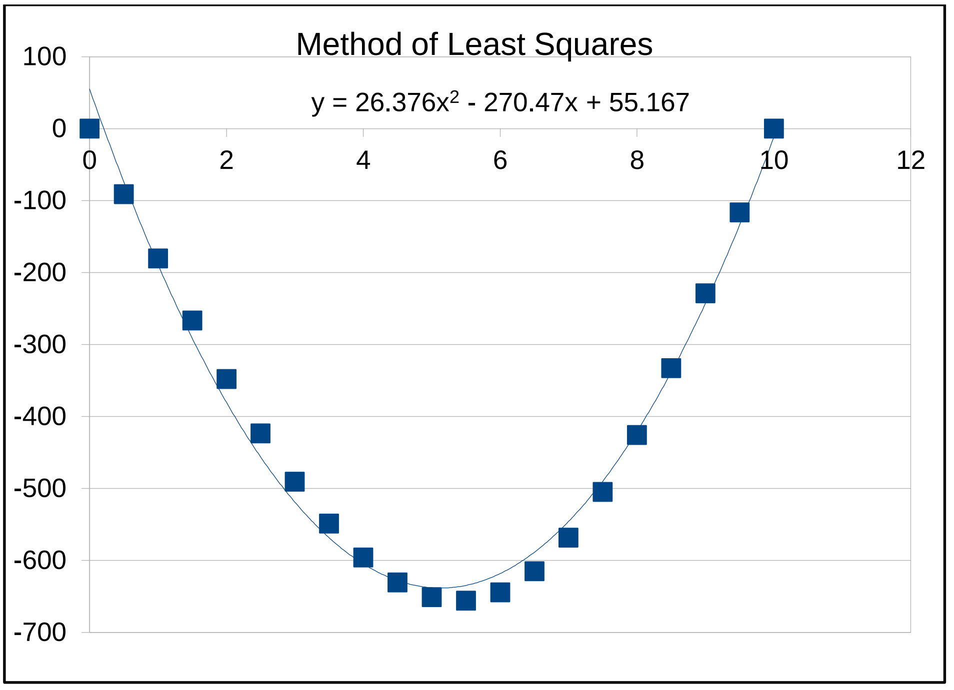

Fig.3.4 showing the second degree deflected curve using MS Excel®

Conclusion

Space for calculations

Date:

EXPERIMENT – 04

<u>TO SOLVE THE SIMULTANEOUS EQUATIONS USING GAUSS ELIMINATION METHOD</u>

Aim

To solve the simultaneous equations using Gauss elimination method.

Tools required

Desktop, Dev C/C++

Theory

Let the given simultaneous equations are given by

$$a_1x + b_1y + c_1z = d_1$$

$$a_2x + b_2y + c_2z = d_2$$

$$a_3x + b_3y + c_3z = d_3$$

Select the pivot row as 1st row, eliminating the first column of 2nd and 3rd row respectively

$$\lambda = \frac{a_{ik}}{a_{kk}}$$

where k is the pivot row = 1, and i is 2nd and 3rd row

Perform the row wise operations, we have $R_2 = R_2 - \lambda R_1$ $R_3 = R_3 - \lambda R_1$

$$a_1x + b_1y + c_1z = d_1$$

$$0x + e_2y + f_2z = g_2$$

$$0x + e_3y + f_3z = g_3$$

where k is the pivot row = 2, and i is 1st and 3rd row $R_1 = R_1 - \lambda R_2$ $R_3 = R_3 - \lambda R_2$

$$a_1x + 0y + h_1z = j_1$$

$$0x + e_2y + f_2z = g_2$$

$$0x + 0y + h_3z = j_3$$

where k is the pivot row = 3, and i is 1st and 2nd row $R_2 = R_2 - \lambda R_3$ $R_1 = R_1 - \lambda R_3$

$$p_1x + 0y + 0z = m_1$$

$$0x + e_2y + 0z = m_2$$

$$0x + 0y + h_3z = j_3$$

Divide the last column with a(i,i) to get the values of x, y, z respectively

Problem statement

Problem 1

The governing equation for the vibration problem is given by $\frac{d^2y}{dx^2} + k^2y = 0$. Using finite differences, find the fundamental frequencies (k). Take the number of nodes N = 5 at an interval of 0.25L. Given y(0) = y(1) = 0. Using power method, calculate the maximum fundamental frequency and the associated mode shape. Take Length L = 1.

Problem 2

The Axial Stiffness matrix of a bar is given as $K = \frac{AE}{L}\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$ and the Force vector is given by $F = \begin{bmatrix} 1000 \\ 1000 \\ -1000 \end{bmatrix} N$. The length 'L' of the bar is equal to 100 mm. Take AE=1. Find the nodal displacements using the relation KU=F, where the nodal displacement vector is given by U = $[U_1\ U_2\ U_3)^T$

Problem 3

The stiffness matrix and the nodal load vector is given below. Solve the simultaneous set of equations and find the nodal displacements.

a.

$$\begin{bmatrix} 409.93597 & -135.1495 & -120.3481 \\ -135.1495 & 195.2655 & -176.9319 \\ -120.3481 & -176.9319 & 121.587 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 1000 \\ 1000 \\ -1000 \end{bmatrix} N$$

(1.243, 1.1327, -5.34529 mm)

b.

$$\begin{bmatrix} 3595.7482 & -1096.608 & -21.8183 \\ -1096.608 & 643.733 & -726.966 \\ -21.8183 & -726.966 & 195.4925 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 1000 \\ 1000 \\ -1000 \end{bmatrix} N$$

(0.5725, 0.9925, -1.3603 mm)

c.

$$\begin{bmatrix} 2786.5697 & 64.8499 & -892.443 \\ 64.8499 & 644.634 & -446.4389 \\ -892.443 & -446.4389 & 214.9102 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 1000 \\ 1000 \\ -1000 \end{bmatrix} N$$

(0.35227, 1.5811, 0.09432 mm)

Problem 4

The flexibility matrix (*f*) and the displacement vector due to the external loading (D_L) for the portal frame is given below. Find the reactions shown as P = $\{P_1, P_2, P_3\}^T$

The Young's Modulus of Elasticity, E = 200 MPa and the moment of Inertia, I = 3.5×10^7 mm^4

EI = 7000 kN-m^4

$$fP + D_L = \frac{1}{6EI}\begin{bmatrix} 750 & 375 & -150 \\ 375 & 2000 & -225 \\ -150 & -225 & 60 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} + \begin{bmatrix} 695/3EI \\ -1390/EI \\ 139/EI \end{bmatrix} = 0$$

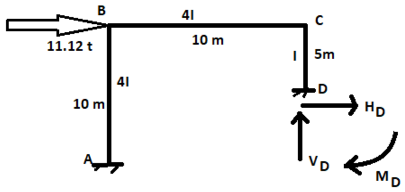

Fig.4.1 A portal frame carrying a horizontal load at the node

(-7.63 kN, 3.27 kN, -20.71 kN-m)

Problem 5

ABCD is a continuous beam having ends A and D fixed at both the ends. The intermediate supports at B and D are simply supported. The span AB is 3m and carries a udl of 40 kN/m. The span CD is 3m and carries a udl of kN/m. Span BC is 2m and carries a point load of magnitude 100kN at a distance of 1m from B. The distance AB=CD= 3m and BC=2m. The support at B settles by y_B= - 2.5 mm (vertically downwards). The stiffness matrix and the nodal load vector are given below. Find the nodal displacements.
The Young's Modulus of Elasticity, E = 200 MPa and the moment of Inertia, I = 3.5 x 10^7 mm^4
EI = 7000 kN-m^2

$$Ku = EI\begin{bmatrix} 35/18 & 5/6 & 6/4 \\ 5/6 & 10/3 & 1 \\ 6/4 & 1 & 10/3 \end{bmatrix}\begin{bmatrix} y_B \\ \theta_B \\ \theta_C \end{bmatrix} = \begin{bmatrix} -110 + R_b \\ 5 \\ -12.5 \end{bmatrix}$$

(- 2.5 mm, 5.096/EI rad, 2.5961/EI rad)

Problem 6

A simply supported beam AB of span 2L carries a point load W acting vertically downwards at the centre of the span. Use the governing equation $EI\frac{d^2y}{dx^2} = M$ and using Finite difference method, find the y-deflection at the centre of the beam. Compare your result with the theoretical value of y-deflection at the centre of the beam. Take EI=W=L=1. Due to symmetry, take five nodes in the half span of the beam at an interval of 0.25L. Given the deflection at the left end A is y(0)=0 and the slope at the mid-point of the beam $y'\left(\frac{2L}{2}\right) = 0$.

Problem 7

Solve the system of equations, $\begin{bmatrix} 4 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ 3 \end{bmatrix}$

Solution by using numerical method namely Gauss Elimination.

Analytical solution/manual solution

Write a program in ‘C’ and verify the solution

Code in ‘C’

Gauss Elimination Method

```
/* Gauss elimination method */
#include <stdio.h>
#define N 3
main()
{

float a[N][N+1], x[N],t,s;
int i,j,k,i1,j1;
printf("Enter the elements of the augmented matrix rowwise\n");
for(i=1;i<=N;i++)
        for(j=1;j<=N+1;j++){
        scanf("%f",&a[i][j]);
        printf( "%8.4f \n", a[i][j]);
        }

for(k=1;k<=N;k++){

        for(i=1;i<=N;i++){

        if(i!=k){
                t = a[i][k]/a[k][k];
                printf(" t = %7.4f \n", t);

                for(j=1;j<=N+1;j++)
                {
                a[i][j]   = a[i][j]-t*a[k][j];
                }

                                for(i1=1;i1<=N;i1++){
```

```
                    for(j1=1;j1<=N+1;j1++){
                    printf( "%8.4f   ", a[i1][j1]);
                    }
                    printf("\n");
                }
        printf("\n");
        }
    }
}

/* now solving */
for (int i = 1; i <=N; i++) printf("%8.4f \n",a[i][N+1]/a[i][i]);

}
```

Conclusions

Inference

Space for calculations

Date:

EXPERIMENT – 05

TO SOLVE THE SIMULTANEOUS EQUATIONS USING GAUSS JORDAN METHOD

Aim

To solve the simultaneous equations using Gauss Jordan method.

Tools required

Desktop, Dev C/C++

Problem Statement

Solve the problems given in Gauss Elimination experiment using Gauss Jordan method

Write a program in ‘C’

Code in ‘C’ for Gauss Jordan Method

```
/* Gauss Jordan method */
#include <stdio.h>
#define N 3
main()
{
float a[N][N+1],t;
int i,j,k;
printf("Enter the elements of the augmented matrix rowwise\n");
for(i=0;i<N;i++)
	for(j=0;j<N+1;j++)
	scanf("%f",&a[i][j]);
/* now calculating the values x1, x2, ... xn */
for(j=0; j<N; j++)
	for(i=0; i<N;i++)
	if(i!=j)
	{
		t = a[i][j]/a[j][j];
		for(k=0;k<N+1;k++)
```

```
                a[i][k]-=a[j][k] * t;
        }
/* now printing the diagonal matrix */
printf("The diagonal matrix is :-\n");
for(i=0;i<N;i++)
{
        for(j=0;j<N+1;j++)
                printf("%9.4f",a[i][j]);
                printf("\n");
}
/* now printing the results */
printf("The solution is :-\n");
for(i=0;i<N;i++)
        printf("x[%3d]=%7.4f\n", i+1,a[i][N]/a[i][i]);
}
```

Conclusions

Date:

EXPERIMENT – 06

TO SOLVE THE SIMULTANEOUS EQUATIONS USING GAUSS SEIDAL METHOD

Aim

To solve the simultaneous equations using Gauss seidal method.

Tools required

Desktop, Dev C/C++

Problem Statement

Solve the problems given in the Gauss Elimination experiment using Gauss Seidal method

Write a Program in 'C'

Code in 'C' for Gauss Seidal Method

```
/* Gauss seidal method */
#include <stdio.h>
#include <math.h>
#define N 3
main()
{
float a[N][N+1], x[N], aerr, maxerr, t, s, err;
int i,j,itr, maxitr;
/* first initialising the array x */
for(i=0;i<N;i++) x[i]=0;
printf("Enter the elements of the augmented matrix rowwise\n");
for(i=0;i<N;i++)
        for(j=0; j<N+1; j++)
                scanf("%f",&a[i][j]);
printf("Enter the allowed error, maximum iterations \n");
scanf("%f %d",&aerr,&maxitr);
printf("Iteration x[1] x[2] x[3]\n");
for(itr=1; itr<=maxitr;itr++)
{
```

```
maxerr=0;
for(i=0;i<N;i++)
{
        s=0;
        for(j=0;j<N;j++)
                if(j!=i) s+=a[i][j]*x[j];
        t = (a[i][N]-s)/a[i][i];
        err = fabs(x[i]-t);
        if(err > maxerr) maxerr = err;
        x[i]=t;
}
printf("%5d",itr);
for(i=0;i<N;i++)
printf("%9.4f",x[i]);
printf("\n");
if(maxerr<aerr)
{
printf("Converges in %3d iterations\n",itr);
for(i=0;i<N;i++)
printf("x[%3d]=%7.4f\n", i+1,x[i]);
return 0;
}
}
printf("Solution does not converge, iterations not sufficient\n");
return 1;
}
```

Conclusions

Space for calculations

Date:

EXPERIMENT – 07

TO FIND THE AREA UNDER THE CURVE USING TRAPEZOIDAL RULE

Aim

To find the slope at the free end of a cantilever using Trapezoidal rule.

Tools required

Desktop, Dev C/C++

Theory

Let the curve be denoted by y = f(x) as shown in the Fig.7.1

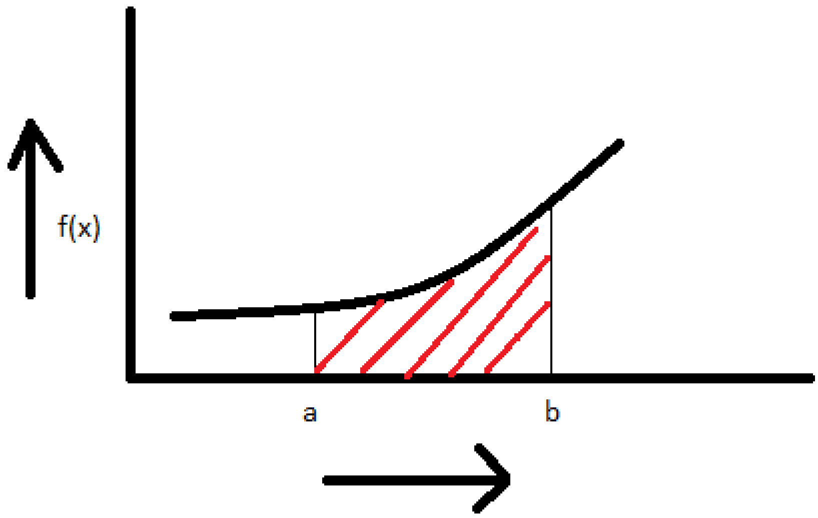

Fig. 7.1 Area under the curve using Trapezoidal rule

Area under the curve = $\frac{h}{2}[f(a) + f(b)],\ \ h = (b - a)$

Problem 1

To find the slope at the free end of the cantilever carrying a point load using Trapezoidal method

A cantilever of length 'L' fixed at left end A and free at the right end B. The beam carries a point load 'W' acting vertically downwards at the free end. Using Moment area theorem – I and trapezoidal rule find the slope at the free end of the cantilever at B. Take the flexural rigidity 'EI' as constant.

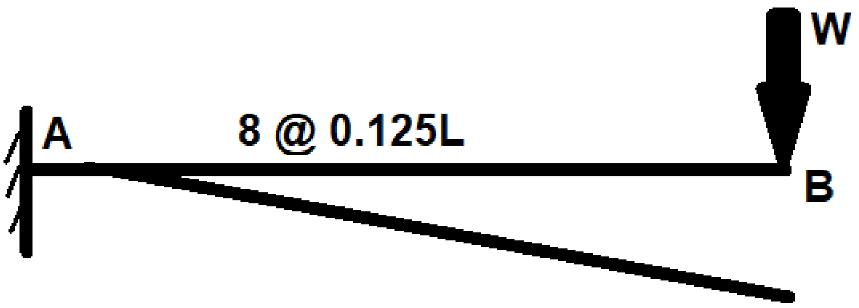

Fig.7.2 A cantilever beam carrying a point load at the free end

Consider a section XX at a distance of x from the free end on the right at B.

$$EI\,\frac{d^2y}{dx^2} = -Wx$$

Integrating, we have

$$EI\frac{dy}{dx} = -\frac{Wx^2}{2} + C_1$$

Integrating again, we have

$$EIy = -\frac{Wx^3}{6} + C_1x + C_2$$

To find the constants of integration, apply the kinematic boundary conditions, we have
At x = l, y=0
At x=l. dy/dx= 0

$$C_1 = \frac{WL^2}{2}$$

$$C_2 = -\frac{WL^3}{3}$$

The equation for the deflected curve is given by

$$EIy = -\frac{Wx^3}{6} + \frac{WL^2}{2}x - \frac{WL^3}{3}$$

Take L = 1, W = 1, EI = 1

Table 1 Deflection and M/EI ordinates for different values of x measured from free end

X	Y	M/EI
0	-1/3	0
0.125L	-833/3072	-0.125
0.25L	-27/128	-0.25
0.375L	-475/3072	-0.375
0.5L	-5/48	-0.5
0.625L	-63/1024	-0.625
0.75L	-11/384	-0.75
0.875L	-0.74869×10^{-3}	-0.875
1.0L	0	-1.0

Using Moment area method, Mohr Theorem – I
The slope at the free end of the cantilever can be calculated as follows.
We draw two tangents one at fixed end on the left at A and the other tangent at the free end on the right at B. The angle between two tangents drawn at A and B should be equal to the area of M/EI diagram between the points A and B.
The ordinates in the M/EI diagram at different values of x (measured from the free end) are given in the table.
The area can be calculated using the Trapezoidal rule.

$$Area = \frac{h}{2}[f(0) + f(n) + 2(f(1) + \cdots + f(n-1)]$$

$$= \frac{-0.125}{2}[(0-1) + 2(-0.125 - 0.25 - 0.375 - 0.5 - 0.625 - 0.75 - 0.875)]$$

$$= \frac{-0.125}{2}\left[-1 - 2 * \frac{7}{2}\right] = \frac{1}{2}$$

The analytical results is

$$\theta_B = \frac{WL^2}{2EI} = \frac{1}{2}$$

Write the program in 'C' and verify your result

Input the values for x as
$x_0 = 0$
$x_n = 1$
sub-intervals is equal to 8

Find the area under the curve using a numerical Integration technique Trapezoidal rule.

Problem 2

To find the slope at the left end support of a simply supported beam carrying a point load at the centre

A simply supported beam AB having length 'L' carries a point load of magnitude 'W' acting vertically downwards at the centre of the span at a distance of 'L/2' from its left end A. Using Moment Area theorem, find the slope at the left end support A. Use (a) Trapezoidal rule and (b) Simpson Rule. Take flexural rigidity as 'EI'.

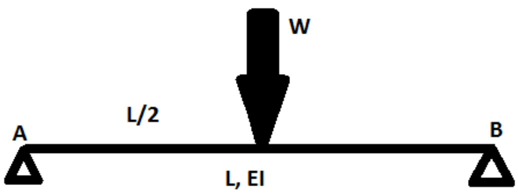

Fig. 7.3 A simply supported beam carrying a point load at the centre

Consider a section XX at a distance of x from the left end support A.

The interval for x is 0 < x < L/2
The vertical reaction at each support is $V_A = V_B = W/2$

$$EI\frac{d^2y}{dx^2} = \frac{W}{2}x$$

Integrating, we have

$$EI\frac{dy}{dx} = \frac{W}{2}\frac{x^2}{2} + C_1$$

$$EI\ y = \frac{W}{2}\frac{x^3}{6} + C_1x + C_2$$

To find the constant of integration, apply the kinematic boundary conditions

At x =0, y = 0 — $C_2 = 0$
At x = L/2, dy/dx = 0 — $C_1 = -WL^2/16$

$$EI\ y = \frac{W}{2}\frac{x^3}{6} - \frac{WL^2}{16}x$$

Take EI = W = L = 1
The ordinates of the deflected curve and M/EI diagram are given in the Table.

X	Y	M/EI
0	0	0
1/16	-191/49152 = - 3.886 x 10^{-3}	1/32
2/16	-47/6144 = 7.649 x 10^{-3}	2/32
3/16	-183/16384 = -0.011169	3/32
4/16	-47/3072 = --0.15299	4/32
5/16	-835/49152 = -0.016988	5/32
6/16	-39/2048 = --0.0190429	6/32
7/16	-1001/49152 = - 0.02036539	7/32
8/16	-1/48 = - 0.0208333	8/32

Apply the Moment area theorem - I
Draw a tangent at A and another tangent at mid span. The angle between two tangents is equal to the area of M/EI diagram between the points.

To find the area of the M/EI diagram we can use Trapezoidal rule / Simpsons rule

Using Trapezoidal rule,

$$\frac{1}{16\ x\ 2}\left[\left(0+\frac{8}{32}\right)+2\left(\frac{1}{32}+\frac{2}{32}+\frac{3}{32}+\frac{4}{32}+\frac{5}{32}+\frac{6}{32}+\frac{7}{32}\right)\right] = \frac{1}{16}$$

Using Simpsons rule,

$$\frac{1}{16\ x\ 3}\left[\left(0+\frac{8}{32}\right)+4\left(\frac{1}{32}+\frac{3}{32}+\frac{5}{32}+\frac{7}{32}\right)+2\left(\frac{2}{32}+\frac{4}{32}+\frac{6}{32}\right)\right] = \frac{1}{16}$$

Analytically, the slope at the left end support is equal to 1/16

Write a program in 'C' and verify your result

Code in 'C'

Take,
x0 =0
xn = 0.5
number of subintervals is equal to 8.

Trapezoidal Rule

```
/* Trapezoidal rule */
#include <stdio.h>
float y(float x)
{
return -x*x*0.5;
}
main()
{
float x0,xn,h,s;
int i,n;
puts("Enter x0, xn,num of subintervals");
scanf("%f %f %d",&x0,&xn,&n);
h=(xn-x0)/n;
s=y(x0)+y(xn);
for(i=1;i<=n-1;i++)
s+=2*y(x0+i*h);
printf("Value of integral is %6.4f\n",(h/2)*s);
}
```

Conclusion

Date:

EXPERIMENT – 08

TO FIND THE AREA UNDER THE CURVE USING SIMPSONS RULE

Aim

To find the slope at the left end support of a simply supported using Simpson's rule.

Tools required

Desktop, Dev C/C++

Theory

Let the curve be denoted by y = f(x) as shown in the Fig.8.1

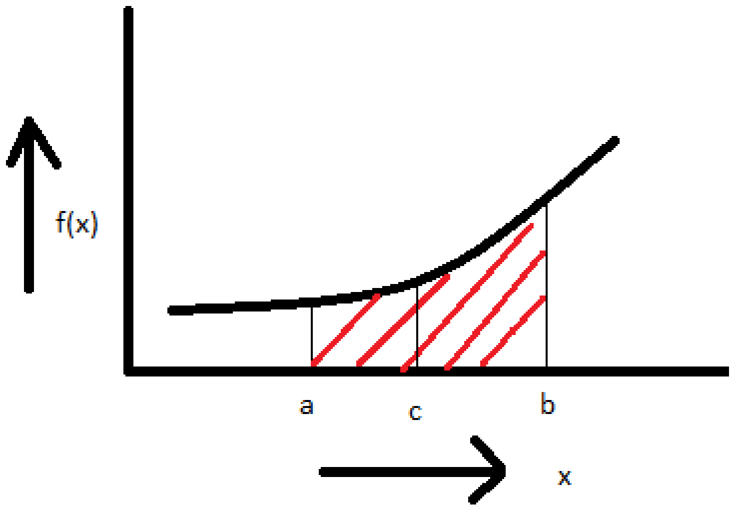

Fig. 8.1 Area under the curve using Simpson's rule

Area under the curve = $\frac{b-a}{6}[(f_0 + f_n) + 4(\sum f_{odd}) + 2(\sum f_{even})]$

Problem statement

Problem 1a

To find the slope at the free end of a cantilever carrying a udl throughout the span using Trapezoidal rule

A cantilever AB is fixed at the left end A and free at the right end B. The beam carries a uniformly distributed load 'w' per unit run over its entire length 'L' as shown in the Fig.8.2. Using Moment area theorem – I and trapezoidal rule find the slope at the free end of the cantilever at B. Take the flexural rigidity 'EI' as constant.

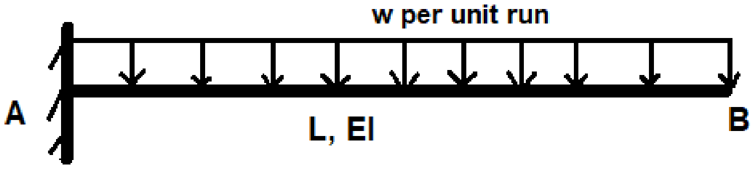

Fig.8.2 A Cantilever beam AB carries udl of intensity ‘w’ per unit run over its entire span

Consider a section XX at a distance of x from the free end on the right at B

Using the second order method, we have

$$EI\frac{d^2y}{dx^2} = -\frac{wx^2}{2}$$

$$EI\frac{dy}{dx} = -\frac{wx^3}{6} + C_1$$

$$EIy = -\frac{wx^4}{24} + C_1x + C_2$$

To find the constants of integration, we apply Kinematic boundary conditions
At x = L y=0
At x = L dy/dx =0

$$C_1 = \frac{wL^3}{6}$$

$$C_2 = -\frac{wL^4}{8}$$

$$EIy == -\frac{wx^4}{24} + \frac{wL^3}{6}x - \frac{wL^4}{8}$$

Take L=1, w=1, EI=1

X	Y	M/EI
0	-0.125	0
0.125	-0.1041768	-1/128
0.25	-0.083496	-4/128
0.375	-0.063323	-9/128
0.5	-0.04427	-16/128
0.625	-0.0271911	-25/128
0.75	-0.0131835	-36/128
0.875	$-3.5909*10^{-3}$	-49/128
1.0	0	-64/128

Moment Area Theorem – I
Draw a tangent at the fixed end on the left at A. Draw a tangent at the free end on the right at B. The angle between the tangents drawn at A and B is equal to the area of the M/EI diagram between A and B.

The area can be determined using the Trapezoidal rule. The ordinates of the M/EI diagram for different values of x measured from the free end on the right at B are given in the table 2.

Area under the curve using the Trapezoidal rule is equal to

$$= \frac{-0.125}{2}\left[\left(0 - \frac{64}{128}\right) + 2\left(-\frac{1}{128} - \frac{4}{128} - \frac{9}{128} - \frac{16}{128} - \frac{25}{128} - \frac{36}{128} - \frac{49}{128}\right)\right]$$

$$= -\frac{0.125}{2}\left[-\frac{64}{128} - \frac{35}{16}\right] = \frac{43}{256} = 0.167968$$

The analytical result is = 0.16666

The M/EI diagram for the cantilever beam carrying udl is a second degree parabola. The linear approximation of this M/EI diagram is a triangle whose area is greater than the area under the parabolic M/EI curve.

Inference

The trapezoidal rule is gives an exact solution when the M/EI diagram is a straight line variation.
When the M/EI diagram is a parabolic variation, the Trapezoidal rule gives an approximate result and over estimates in this case of a cantilever beam carrying a udl over its entire span.
To improve the accuracy of the result the number of intervals can be increased significantly. The computational effort required shall be higher as well.
Simpson's rule can be used instead of a Trapezoidal rule to determine the slope at the free end of a cantilever carrying a udl over its entire span. The variation of the M/EI diagram is parabolic in nature.

Write the program in 'C' and verify your result

Take initial value of x0 =0
xn = 1
number of intervals = 8
number of intervals = 1000

Problem 1b

To find the slope at the free end of a cantilever carrying udl throughout the span using Simpsons Rule

A cantilever AB is fixed at the left end A and free at the right end B. The beam carries a uniformly distributed load 'w' per unit run over its entire length 'L' as shown in the Fig.8.3. Using Moment area theorem – I and trapezoidal rule find the slope at the free end of the cantilever at B. Take the flexural rigidity 'EI' as constant.

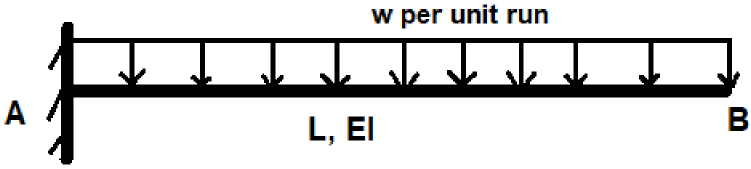

Fig.8.3 A Cantilever beam AB carries udl of intensity 'w' per unit run over its entire span

Consider a section XX at a distance of x from the free end on the right at B

Using the second order method, we have

$$EI\frac{d^2y}{dx^2} = -\frac{wx^2}{2}$$

$$EI\frac{dy}{dx} = -\frac{wx^3}{6} + C_1$$

$$EIy = -\frac{wx^4}{24} + C_1x + C_2$$

To find the constants of integration, we apply Kinematic boundary conditions
At x = L y=0
At x = L dy/dx =0

$$C_1 = \frac{wL^3}{6}$$

$$C_2 = -\frac{wL^4}{8}$$

$$EIy == -\frac{wx^4}{24} + \frac{wL^3}{6}x - \frac{wL^4}{8}$$

Take L=1, w=1, EI=1

X	Y	M/EI
0	-0.125	0
0.125	-0.1041768	-1/128
0.25	-0.083496	-4/128
0.375	-0.063323	-9/128
0.5	-0.04427	-16/128
0.625	-0.0271911	-25/128

0.75	-0.0131835	-36/128
0.875	$-3.5909*10^{-3}$	-49/128
1.0	0	-64/128

Moment Area Theorem – I

Draw a tangent at the fixed end on the left at A. Draw a tangent at the free end on the right at B. The angle between the tangents drawn at A and B is equal to the area of the M/EI diagram between A and B. The area can be determined using the Simpsons rule. The ordinates of the M/EI diagram for different values of x measured from the free end on the right at B are given in the table 3.

Area under the curve using the Simpsons rule is equal to

$$= \frac{-0.125}{3}\left[\left(0 - \frac{64}{128}\right) + 4\left(-\frac{1}{128} - \frac{9}{128} - \frac{25}{128} - \frac{49}{128}\right) + 2\left(-\frac{4}{128} - \frac{16}{128} - \frac{36}{128}\right)\right] = \frac{1}{6}$$

The analytical result is = 0.1666

Write the program in ‘C’ to determine the slope at the free end of a Cantilever using Simpson’s rule

X0 = 0
Xn = 1
Number of subintervals is equal to 8.

Problem 2

To find the slope at the left end support of a simply supported beam carrying a point load at the centre

A simply supported beam AB having length 'L' carries a point load of magnitude 'W' acting vertically downwards at the centre of the span at a distance of 'L/2' from its left end A. Using Moment Area theorem, find the slope at the left end support A. Use (a) Trapezoidal rule and (b) Simpson Rule. Take flexural rigidity as 'EI'.

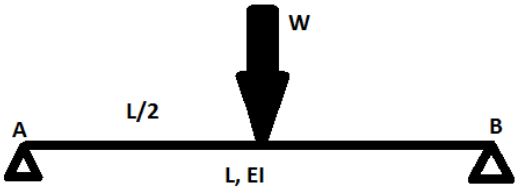

Fig.8.4 A simply supported beam carrying a point load W at the centre

Consider a section XX at a distance of x from the left end support A.
The interval for x is $0 < x < L/2$
The vertical reaction at each support is $V_A = V_B = W/2$

$$EI\frac{d^2y}{dx^2} = \frac{W}{2}x$$

Integrating, we have

$$EI\frac{dy}{dx} = \frac{W}{2}\frac{x^2}{2} + C_1$$

$$EI\ y = \frac{W}{2}\frac{x^3}{6} + C_1x + C_2$$

To find the constant of integration, apply the kinematic boundary conditions

At x =0, y = 0 $\qquad$ $C_2 = 0$
At x = L/2, dy/dx = 0 $\qquad$ $C_1 = - WL^2/16$

$$EI\ y = \frac{W}{2}\frac{x^3}{6} - \frac{WL^2}{16}x$$

Take EI = W = L = 1
The ordinates of the deflected curve and M/EI diagram are given in the Table.

X	Y	M/EI
0	0	0
1/16	-191/49152 = - 3.886 x 10^{-3}	1/32
2/16	-47/6144 = 7.649 x 10^{-3}	2/32
3/16	-183/16384 = -0.011169	3/32
4/16	-47/3072 = --0.15299	4/32
5/16	-835/49152 = -0.016988	5/32

6/16	-39/2048 = --0.0190429	6/32
7/16	-1001/49152 = -0.02036539	7/32
8/16	-1/48 = - 0.0208333	8/32

Apply the Moment area theorem - I
Draw a tangent at A and another tangent at mid span. The angle between two tangents is equal to the area of M/EI diagram between the points.

To find the area of the M/EI diagram we can use Trapezoidal rule / Simpsons rule

Using Trapezoidal rule,

$$\frac{1}{16\,x\,2}\left[\left(0+\frac{8}{32}\right)+2\left(\frac{1}{32}+\frac{2}{32}+\frac{3}{32}+\frac{4}{32}+\frac{5}{32}+\frac{6}{32}+\frac{7}{32}\right)\right]=\frac{1}{16}$$

Using Simpsons rule,

$$\frac{1}{16\,x\,3}\left[\left(0+\frac{8}{32}\right)+4\left(\frac{1}{32}+\frac{3}{32}+\frac{5}{32}+\frac{7}{32}\right)+2\left(\frac{2}{32}+\frac{4}{32}+\frac{6}{32}\right)\right]=\frac{1}{16}$$

Analytically, the slope at the left end support is equal to 1/16

Write a program in ‘C’ and verify your result
Take,
x0 =0
xn = 0.5
number of subintervals is equal to 8.

Trapezoidal Rule

Simpsons Rule

Problem 3

To find the slope at the left end support of a simply supported beam carrying udl throughout the span

A simply supported beam AB of span 'L' and flexural rigidity 'EI' carries a uniformly distributed load of intensity 'w' per unit run over the entire span. Find the slope at the left end support using (a) Trapezoidal Rule and (b) Simpson's rule
Write a program in 'C' and verify your results.

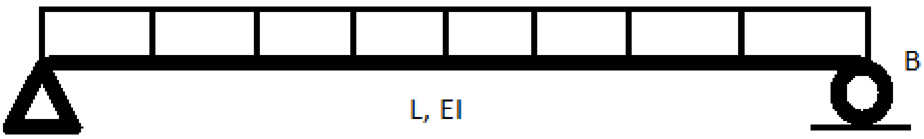

Fig.8.5 A simply supported beam carrying a udl throughout its entire span

Consider a section XX at a distance of x from the left end support at A.
The vertical reactions at each support $V_A = V_B = wL/2$

Using the second order method, we have

$$EI\,\frac{d^2y}{dx^2} = \frac{wL}{2}x - \frac{wx^2}{2}$$

Integrating, we have

$$EI\frac{dy}{dx} = \frac{wL}{2}\frac{x^2}{2} - \frac{w}{2}\frac{x^3}{3} + C_1$$

$$EI\;y = \frac{wL}{2}\frac{x^3}{6} - \frac{\frac{w}{2}x^4}{12} + C_1x + C_2$$

Take, w = L = EI = 1
To find C_1 and C_2, apply the kinematic boundary conditions
At x = 0 y = 0 $\quad C_2 = 0$
At x = L/2 dy/dx = 0 $\quad C_1 = -1/24$

$$EI\;y = \frac{wL}{2}\frac{x^3}{6} - \frac{\frac{w}{2}x^4}{12} - \frac{wL^3x}{24}$$

The values for the ordinates in the deflected curve and the M/EI diagram are given in the table

X	Y	M/EI
0	0	0
1/16	-2.58445×10^{-3}	15/512
2/16	$-497/98304 = -5.0557 \times 10^{-3}$	7/128=28/512
3/16	-7.31468×10^{-3}	39/512
4/16	$-19/2048 = -9.277\ 10^{-3}$	3/32=48/512
5/16	-0.010875	55/512
6/16	-395/32768=-0.01205	15/128=60/512
7/16	-0.0127777	63/512
8/16	-5/384=-0.0130208	1/8=64/512

Moment Area Theorem - I

Draw two tangents at left end A and at mid span. The angle between the tangents is equal to the area of the M/EI diagram. The slope at the left end support is equal to the area of the M/EI diagram between the tangents.

To find the area under the curve use Trapezoidal rule and Simpson's rule

Trapezoidal Rule

$$= \frac{1}{2} * \frac{1}{16}\left[\left(0 + \frac{64}{512}\right) + 2\left(\frac{15}{512} + \frac{28}{512} + \frac{39}{512} + \frac{48}{512} + \frac{55}{512} + \frac{60}{512} + \frac{63}{512}\right)\right] = \frac{85}{2048}$$

Simpsons Rule

$$= \frac{1}{16} * \frac{1}{3}\left[\left(0 + \frac{64}{512}\right) + 4\left(\frac{15}{512} + \frac{39}{512} + \frac{55}{512} + \frac{63}{512}\right) + 2\left(\frac{28}{512} + \frac{48}{512} + \frac{60}{512}\right)\right] = \frac{1}{24}$$

Analytical Result = 1/24

The trapezoidal method uses a linear approximation of the M/EI curve. To increase the accuracy, increase the number of subintervals.

Simpsons rule is more accurate for parabolic curves.

Write a program in 'C' and verify your results

Take

X0 = 0

Xn = 0.5

Number of subintervals for **Trapezoidal rule**

8 and 1000

Number of subintervals for **Simpsons rule** is 8

Write a Program in ‘C’

Code in ‘C’

```
/* Simpsons rule */
#include <stdio.h>
float y(float x)
{
return -x*x*0.5;
}
main()
{
float x0,xn,h,s;
int i,n;
puts("Enter x0,xn,num of subintervals");
scanf("%f %f %d",&x0,&xn,&n);
h = (xn-x0)/n;
s=y(x0)+y(xn)+4*y(x0+h);
for(i=3;i<=n-1;i+=2)
s+=4*y(x0+h*i)+2*y(x0+(i-1)*h);
printf("Value of integral is %6.4f\n",(h/3)*s);
}
```

Conclusion

Date:

EXPERIMENT – 09

BUCKLING OF COLUMNS

Aim
To find the deflection of a laterally loaded column using Euler's method.

Tools required
Desktop, Dev C/C++

Theory

The laterally loaded column is as shown in the Fig.9.1.

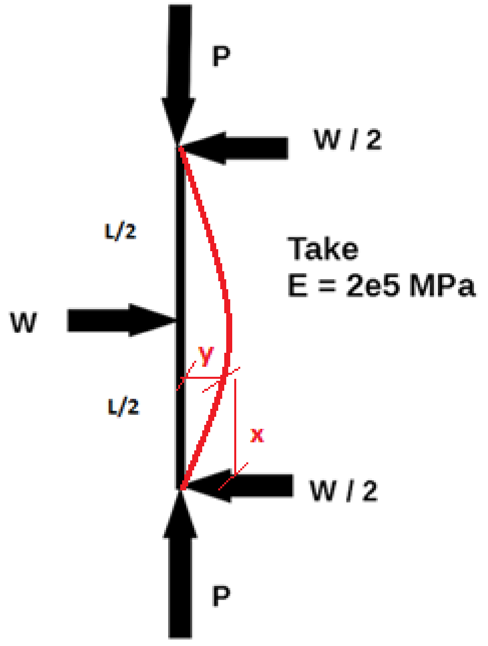

Fig.9.1 Laterally loaded column

The governing equation is given by

$$EI\,\frac{d^2y}{dx^2} = -\frac{W}{2}x - Py$$

The solution to the above differential equation is of the form

$$y = C_1 cosx\sqrt{\frac{P}{EI}} + C_2 sin\sqrt{\frac{P}{EI}}x - \frac{Wx}{2P}$$

The slope at the section in AC is given by

$$\frac{dy}{dx} = -C_1\sqrt{\frac{P}{EI}}sinx\sqrt{\frac{P}{EI}} + C_2\sqrt{\frac{P}{EI}}cosx\sqrt{\frac{P}{EI}} - \frac{W}{2P}$$

At x= 0 y = 0 $C_1 = 0$

At x = l/2 dy/dx = 0 $C_2 = \frac{W}{2P}\sqrt{\frac{P}{EI}}\sec\left(\frac{l}{2}\sqrt{\frac{P}{EI}}\right)$

The deflection at any section in first half of the span is given

$$y = \frac{W}{2P}\sqrt{\frac{EI}{P}}\sec\left(\left(\frac{l}{2}\right)\sqrt{\frac{P}{EI}}\right) sinx\sqrt{\frac{P}{EI}} - \frac{Wx}{2P}$$

$$y_{max} = \frac{W}{2P}\sqrt{\frac{EI}{P}}\tan\left(\left(\frac{l}{2}\right)\sqrt{\frac{P}{EI}}\right) - \frac{Wl}{4P}$$

Bending moment maximum = $-Py_{max} - \frac{Wl}{4} = -\frac{W}{2}\sqrt{\frac{EI}{P}}\tan\left(\left(\frac{l}{2}\right)\sqrt{\frac{P}{EI}}\right)$

Limiting maximum compressive stress is say, 150 MPa i.e., 0.15 KN/mm^2

$$\frac{P}{A} \pm \frac{M_{max}}{I} = 0.15$$

Problem Statement

A laterally loaded column of steel 50 mm x 50 mm in section and 2m long carries an axial load of 100 kN and a lateral load of W = 3014 N at the centre normal to one of the faces as shown in the Fig.9.2 below. The column is hinged at both the ends. Take E = 2e5 MPa. Find the maximum deflection using Euler's method to solve the linear differential equation as given below. Plot the deflected shape of the curve. Also, find the maximum bending moment.

The governing differential equation of the column is given by

$$\frac{dy}{dx} = (W/2\,P)\cos\left(x\sqrt{\frac{P}{EI}}\right)/\cos\left(\frac{L}{2}\sqrt{\frac{P}{EI}}\right) - \frac{W}{2P}$$ where x lies in the interval of [0m 1m]

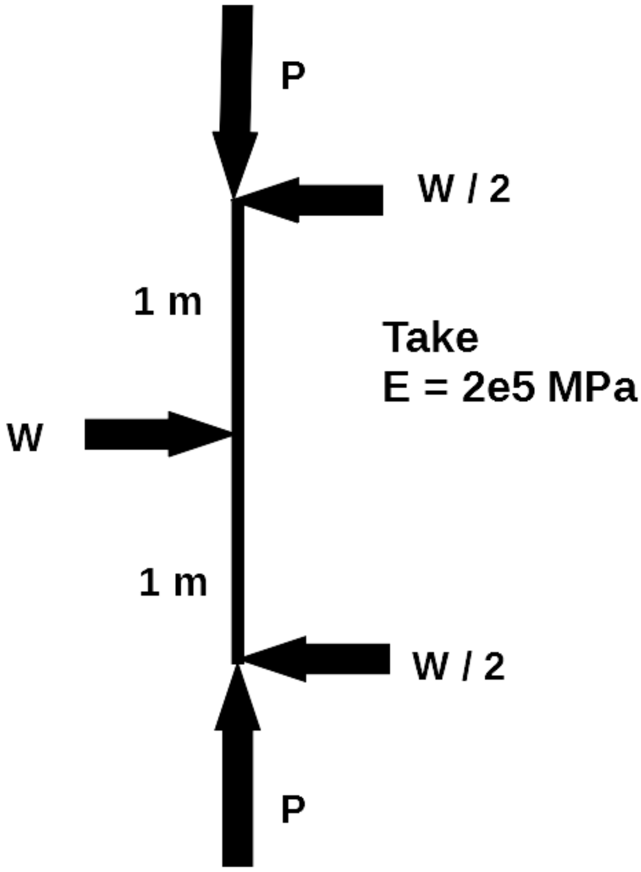

Fig. 9.2 Laterally Loaded column

Analytical Method

Lateral displacement

Maximum bending moment

Write a program in 'C' for Euler's method

Code in in C

```
/* Buckling of Laterally loaded columns using Euler's Method */
#include <stdio.h>
#include <math.h>
float df(float x, float y)
{
	float w,l,p,sqrtpbyei,ei,term1;
	w = 3014;
	l = 2000.0;
	p = 100e3;
	ei = 2e5*520833.333;
	sqrtpbyei = sqrt(p/ei);/*)288.675e-6;*/
	term1 = w/(2.0*p);
	term1 = term1*cos(x*sqrtpbyei)/cos(l*0.5*sqrtpbyei);
	term1 = term1 - 0.5*(w/p);
	return term1;
}
int main()
{
	float x0,y0,h,x,x1,y1;
	puts("Enter the values of x0 y0 h x");
	scanf("%f %f %f %f", &x0, &y0, &h, &x);

	x1=x0; y1=y0;
	while(1)
	{
		if(x1>=x) return 0;
		y1+=h*df(x1,y1);
		x1+=h;
//		printf("When x = %3.1f  y = %4.2f\n", x1, y1);
		printf("%8.4f				%8.4f\n",x1,y1);
	}
}
```

Maximum deflection of the laterally loaded column using the ‘C’ program is equal to 7.91 mm.

Table 1 showing the deflection in mm along the length of the column

δ (mm)	L (mm)	δ (mm)	L (mm)	δ (mm)	L (mm)	δ (mm)	L (mm)
0.1198	10	5.5574	510	7.9095	1010	5.3803	1510
0.2395	20	5.6441	520	7.9051	1020	5.2899	1520
0.3592	30	5.7296	530	7.8985	1030	5.1983	1530
0.4789	40	5.8137	540	7.8899	1040	5.1056	1540
0.5984	50	5.8964	550	7.879	1050	5.0117	1550
0.7179	60	5.9779	560	7.8661	1060	4.9167	1560
0.8372	70	6.0579	570	7.8511	1070	4.8205	1570
0.9563	80	6.1366	580	7.834	1080	4.7234	1580
1.0752	90	6.2139	590	7.8148	1090	4.6251	1590
1.1939	100	6.2897	600	7.7935	1100	4.5259	1600
1.3124	110	6.364	610	7.7702	1110	4.4256	1610
1.4306	120	6.4369	620	7.7449	1120	4.3244	1620
1.5485	130	6.5083	630	7.7176	1130	4.2222	1630
1.6661	140	6.5781	640	7.6883	1140	4.1191	1640
1.7833	150	6.6464	650	7.6571	1150	4.0151	1650
1.9001	160	6.7132	660	7.6238	1160	3.9102	1660
2.0166	170	6.7783	670	7.5887	1170	3.8044	1670
2.1326	180	6.8419	680	7.5516	1180	3.6978	1680
2.2482	190	6.9038	690	7.5127	1190	3.5905	1690
2.3633	200	6.9641	700	7.4718	1200	3.4823	1700
2.4778	210	7.0227	710	7.4291	1210	3.3734	1710
2.5919	220	7.0796	720	7.3846	1220	3.2637	1720
2.7054	230	7.1348	730	7.3382	1230	3.1534	1730
2.8183	240	7.1883	740	7.29	1240	3.0423	1740
2.9307	250	7.24	750	7.24	1250	2.9307	1750
3.0423	260	7.29	760	7.1883	1260	2.8183	1760
3.1534	270	7.3382	770	7.1348	1270	2.7054	1770
3.2637	280	7.3846	780	7.0796	1280	2.5919	1780
3.3734	290	7.4291	790	7.0227	1290	2.4778	1790
3.4823	300	7.4718	800	6.9641	1300	2.3633	1800
3.5905	310	7.5127	810	6.9038	1310	2.2482	1810
3.6978	320	7.5516	820	6.8419	1320	2.1326	1820
3.8044	330	7.5887	830	6.7783	1330	2.0166	1830
3.9102	340	7.6238	840	6.7132	1340	1.9001	1840
4.0151	350	7.6571	850	6.6464	1350	1.7833	1850
4.1191	360	7.6883	860	6.5781	1360	1.6661	1860
4.2222	370	7.7176	870	6.5083	1370	1.5485	1870
4.3244	380	7.7449	880	6.4369	1380	1.4306	1880
4.4256	390	7.7702	890	6.364	1390	1.3124	1890
4.5259	400	7.7935	900	6.2897	1400	1.1939	1900
4.6251	410	7.8148	910	6.2139	1410	1.0752	1910
4.7234	420	7.834	920	6.1366	1420	0.9563	1920
4.8205	430	7.8511	930	6.0579	1430	0.8372	1930
4.9167	440	7.8661	940	5.9779	1440	0.7179	1940
5.0117	450	7.879	950	5.8964	1450	0.5984	1950
5.1056	460	7.8899	960	5.8137	1460	0.4789	1960
5.1983	470	7.8985	970	5.7296	1470	0.3592	1970
5.2899	480	7.9051	980	5.6441	1480	0.2395	1980
5.3803	490	7.9095	990	5.5574	1490	0.1198	1990
5.4695	500	7.9116	1000	5.4695	1500	0	2000

Deflected shape of the laterally loaded column using Euler's method

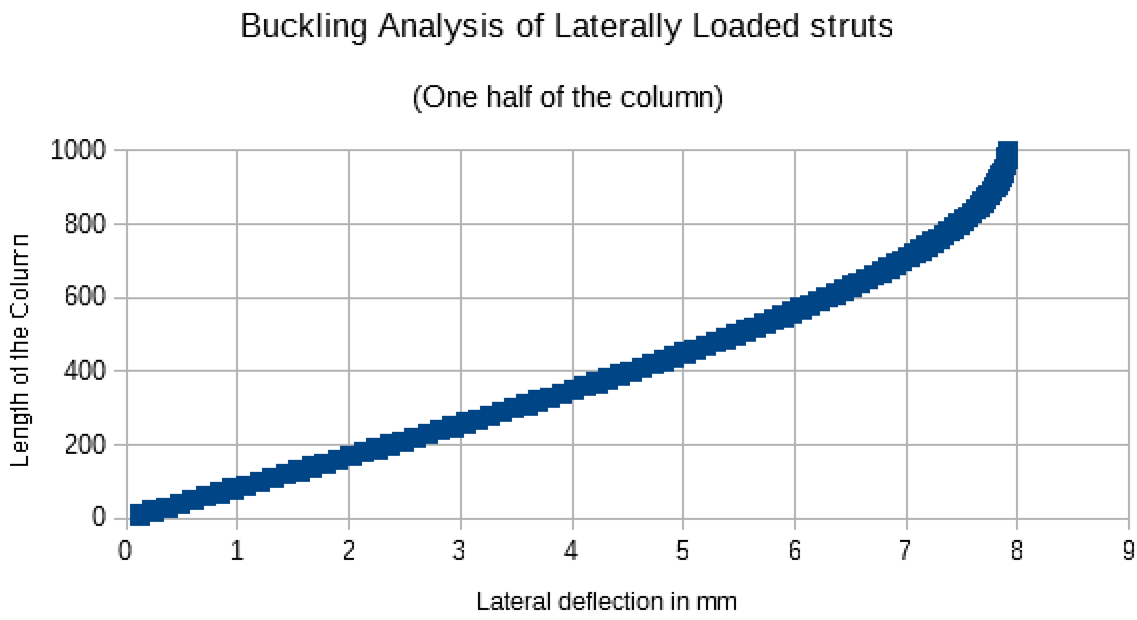

Fig.9.3 showing the deflected curve of the one half of the column using Euler's method

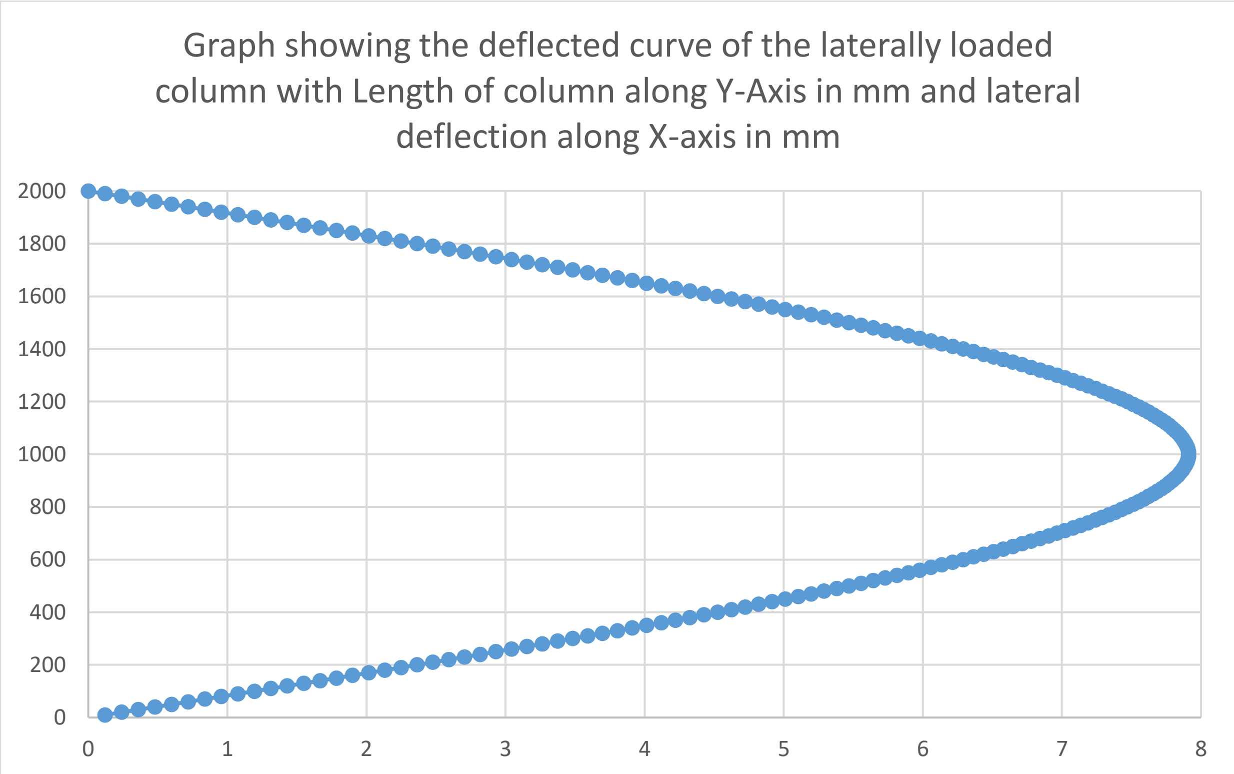

Fig.9.4 showing the deflected curve of the laterally loaded column using Euler's method (By symmetry)

The maximum deflection of the laterally loaded column using Euler's method is equal to 7.9116 mm.

Screen shot of the output of the program in ‘C’

```
750.0000                                  7.2400
760.0000                                  7.2900
770.0000                                  7.3382
780.0000                                  7.3846
790.0000                                  7.4291
800.0000                                  7.4718
810.0000                                  7.5127
820.0000                                  7.5516
830.0000                                  7.5887
840.0000                                  7.6238
850.0000                                  7.6571
860.0000                                  7.6883
870.0000                                  7.7176
880.0000                                  7.7449
890.0000                                  7.7702
900.0000                                  7.7935
910.0000                                  7.8148
920.0000                                  7.8340
930.0000                                  7.8511
940.0000                                  7.8661
950.0000                                  7.8790
960.0000                                  7.8899
970.0000                                  7.8985
980.0000                                  7.9051
990.0000                                  7.9095
1000.0000                                 7.9116

--------------------------------
Process exited with return value 0
Press any key to continue . . .
```

Maximum Bending Moment is equal to

Conclusion

Space for calculations

Date:

EXPERIMENT – 10

RUNGA KUTTA METHOD

Aim

To find the solution of a second order differential equation using Runga Kutta method

Tools required

Desktop, Dev C/C++

Theory

$$\frac{dy}{dx} = f(x,y,z) \; and \; \frac{dz}{dx} = \phi(x,y,z)$$

With the initial conditions $y(x_0) = y_0$ and $z(x_0)=z_0$ can be solved by applying Runge Kutta method Starting at (x_0,y_0,z_0) and taking the step sizes for x, y, z to be h,k,l respectively, the Runge Kutta Method gives,

$k_1 = hf(x_0, y_0, z_0)$	$l_1 = h\phi(x_0, y_0, z_0)$
$k_2 = hf(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_1, z_0 + \frac{1}{2}l_1)$	$l_2 = h\phi(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_1, z_0 + \frac{1}{2}l_1)$
$k_3 = hf(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_2, z_0 + \frac{1}{2}l_2)$	$l_3 = h\phi(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_2, z_0 + \frac{1}{2}l_2)$
$k_4 = hf(x_0 + h, y_0 + k_3, z_0 + l_3)$	$l_4 = h\phi(x_0 + h, y_0 + k_3, z_0 + l_3)$
$y_1 = y_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$	$z_1 = z_0 + \frac{1}{6}(l_1 + 2l_2 + 2l_3 + l_4)$

Problem Statement

The differential equation is given by $y'' + xy' + y = 0, y(0) = 1, y'(0) = 0$

Obtain y for x = 0 (0.1) 0.1 by using Runge Kutta method. h = 0.1

Substitute $y' = z$, the given equation reduces to the simultaneous equations

$$z' + xz + y = 0 \; and \; y' = z$$

$$\frac{dy}{dx} = z = f(x,y,z)$$

$$\frac{dz}{dx} = \phi(x,y,z) = -xz - y$$

$$x_0 = 0, y_0 = 1, z_0 = 0, y'(0) = 0, h = 0.1$$

Runga kutta becomes

$k_1 = hf(x_0, y_0, z_0) = 0.1(0) = 0$

$k_2 = hf\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_1, z_0 + \frac{1}{2}l_1\right)$

$l_1 = h\phi(x_0, y_0, z_0) = 0.1(-1) = -0.1$

$l_2 = h\phi(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_1, z_0 + \frac{1}{2}l_1)$

$k_3 = hf\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_2, z_0 + \frac{1}{2}l_2\right)$

$l_3 = h\phi(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_2, z_0 + \frac{1}{2}l_2)$

$k_4 = hf(x_0 + h, y_0 + k_3, z_0 + l_3)$

$l_4 = h\phi(x_0 + h, y_0 + k_3, z_0 + l_3)$

$y_1 = y_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$

$z_1 = z_0 + \frac{1}{6}(l_1 + 2l_2 + 2l_3 + l_4)$

Solution using Runga Kutta method (Manually)

Write a Program in C

Code in ‘C’

```
/*Runge Kutta method */
#include <stdio.h>
float f(float x, float y, float z)
{
return z;
}
float phi(float x, float y, float z)
{
return -x*f(x,y,z)-y;
}

main()
{
float x0,y0,z0,h,xn,x,y,z,k1,k2,k3,k4,k,l1,l2,l3,l4,l;
int n,N;
//printf("Enter the value of x0,y0,z0,h,xn \n");
//scanf("%f %f %f %f",&x0,&y0,&h,&xn);
x0=0;y0=1;h=0.1;xn=0.5;
x=x0; y=y0;
for(n=1;n<=10;n++)
{
if(x==xn) break;
k1 = h*f(x,y,z);
l1 = h*phi(x,y,z);
k2 = h*f(x+h/2, y+k1/2, z+l1/2);
l2 = h*phi(x+h/2, y+k1/2, z+l1/2);
k3 = h*f(x+h/2, y+k2/2, z+l2/2);
l3 = h*phi(x+h/2, y+k2/2, z+l2/2);
k4 = h*f(x+h, y+k3, z+l3);
l4 = h*phi(x+h, y+k3, z+l3);
k=(k1+k2*2+k3*2+k4)/6;
```

```
l = (l1+l2*2+l3*2+l4)/6;
x+=h;y+=k;z+=l;
printf(" k1=%8.4f k2=%8.4f k3=%8.4f k4=%8.4f\n",k1,k2,k3,k4);
printf(" l1=%8.4f l2=%8.4f l3=%8.4f l4=%8.4f\n",l1,l2,l3,l4);

printf("When x = %8.4f y=%8.4f z=%8.4f\n",x,y,z);
}
}
```

Solution using the ‘C’ program

Conclusion

Space for calculations