

Name - Deepankar Sharma

Course - BCA

Student ID - 20041299

University Roll No - 2092014

CBNST Fnd Term Practical PBC-302

/* Bisection Method */

Ques-1- #include <stdio.h>

#include <math.h>

#define f(x) x*x*x - 6

int main () {

int iter = 1;

float a, b, c, allowr;

printf ("Enter the value of a (negative): ");

scanf ("%f", &a);

printf ("Enter the value of b (positive): ");

scanf ("%f", &b);

printf ("Enter the value of allowed error: ");

scanf ("%f", &allowr);

printf ("a \t b \t f(a) \t f(b) \t c \t f(c) \n");

do {

c = (a+b)/2;

printf ("%d \t %d \t %f \t %f \t %f \t %f \n", a, b, f(a), f(b), c, f(c));

if (f(c) < 0) { a = c; }

~~else~~ else { b = c; }

} while (fabs(f(c)) > allowr);

printf ("\n\n The root is : %f \n", c);

return 0;

}

* Output */

Enter the value of a (negative): -1.5

Enter the value of b (positive): 1.9

Enter the value of allowed error: 0.01

a	b	f(a)	f(b)	c	f(c)
-1.50000	1.90000	-2.62500	0.859000	1.7000	-1.087000
1.70000	1.90000	-1.08700	0.859000	1.8000	-0.168001
1.80000	1.90000	-0.168001	0.859000	1.85000	0.331624
1.80000	1.85000	-0.168001	0.331624	1.825000	-0.01483
1.80000	1.825000	-0.168001	0.331624	1.812500	0.000662
1.812500	1.818750	-0.045654	0.016154	1.818750	-0.045654
1.812500	1.818750	-0.045654	0.016154	1.812500	-0.045654

The root is: ~~1.818750~~ 1.812500

Ques-3

PAGE NO.:
DATE: / / 20PAGE NO.:
DATE: / / 20

/* Lagrange Method */

#include <stdio.h>

#include <math.h>

int main()

int n;

printf("Enter the number of unknown variables: ");

scanf("%d", &n);

static float mat[10][2] = {0.0};

for (int i=0; i<n; i++)

printf("Enter x%d: ", i+1);

scanf("%f", &mat[i][0]);

printf("Enter y%d: ", i+1);

scanf("%f", &mat[i][1]);

}

// printing table

for (int i=0; i<n; i++)

printf("%f \t %f \n", mat[i][0], mat[i][1]);

}

float x, sum = 0.0;

printf("Enter x: "); scanf("%f", &x);

for (int i=0; i<n; i++)

float z = 1.0;

for (int j=0; j<n; j++)

if (i!=j)

z *= (x - mat[j][0]);

z /= (mat[i][0] - mat[j][0]);

}

z *= mat[i][1]; sum += z;

}

printf("The value of f(%f) is %f.", x, sum); return 0;

}

/* Output */

Enter the number of unknown variables: 4

Enter x1: 4

Enter y1: 20

Enter x2: 5

Enter y2: 25

Enter x3: 6

Enter y3: 7

Enter x4: 3

Enter y4: 54

4 20

5 25

6 7

3 54

Enter x: 5

The value of f(5.00000) is 25.00000

ques 4.

/* Gauss Elimination Method */

#include <stdio.h>

int main()

```

    int n; printf("Enter unknown: ");
    scanf("%d", &n);
    float arr[n][n+1], x[n];
    float X[n]; printf("Enter values:\n");
    for (int i=0; i<n; i++) {
        for (int j=0; j<n; j++) {
            scanf("%f", &arr[i][j]);
        }
    }

```

{ // upper triangular matrix

```

    for (int i=0; i<n-1; i++) {
        for (int j = i+1; j<n; j++) {
            float p = arr[j][i] / arr[i][i];
            for (int k=0; k<n+1; k++) {
                arr[j][k] = arr[j][k] - p * arr[i][k];
            }
        }
    }

```

{ // backward substitution

for (int i=n-1; i>=0; i--)

double sum=0;

for (int j = i+1; j<n; j++)

sum = sum + arr[i][j] * X[j];

} X[i] = (arr[i][n] - sum) / arr[i][i];

{

printf("The values of unknown variables is: \n");

for (int i=0; i<n; i++)

printf("X[%d] = %f \n", i, X[i]);

} return 0;
}

/* Output */

Enter unknown : 3

Enter values :

4

6

8

4

7

4

6

3

5

3

5

6

The values of unknown variables is :

$X[0] = -1$

$X[1] = -8$

$X[2] = 7$