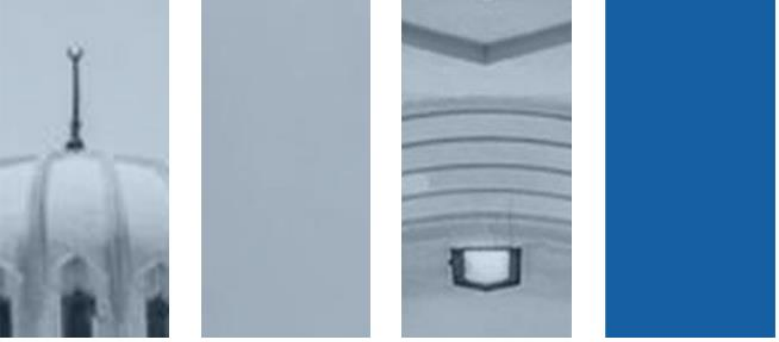


# RESEARCH & PROJECT SUBMISSIONS

**Program:** **MCTA**

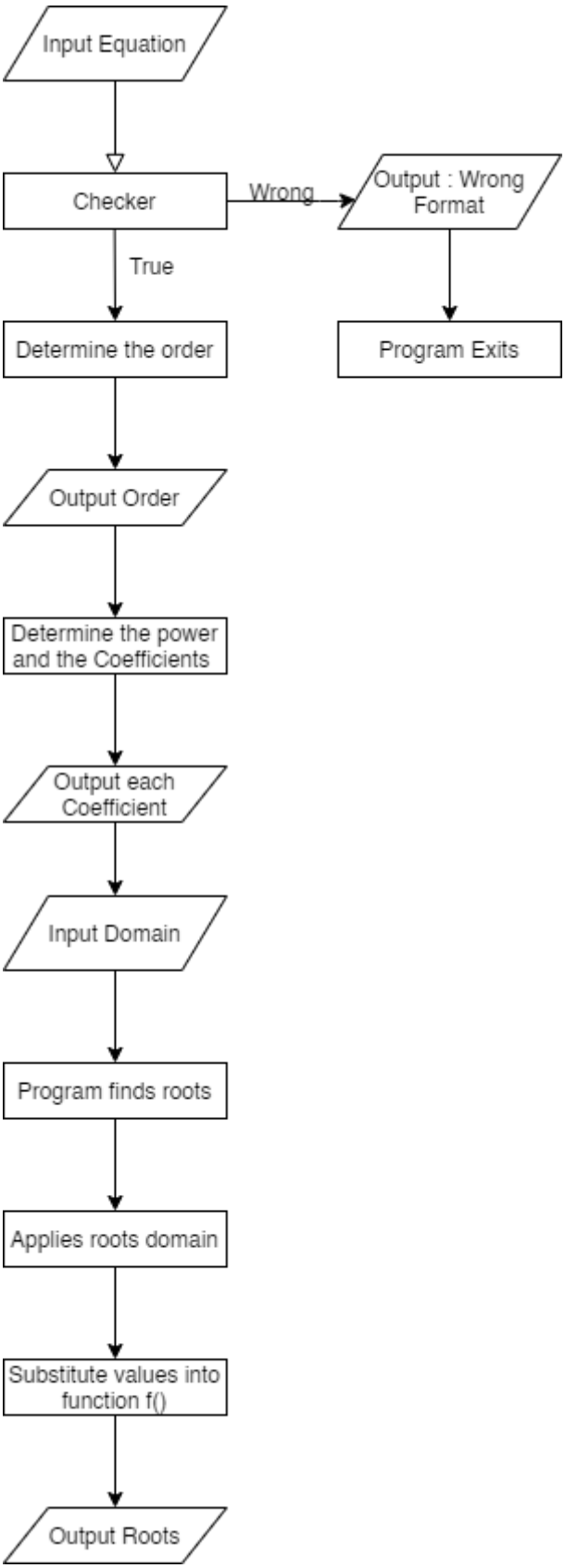
*Course Code:* ***CSE 131***

*Course Name:* ***Computer Programming***

*Examination Committee*

**Ain Shams University**
**Faculty of Engineering**
**International Credit Hours Engineering Programs (I-CHEP)**

## Program Theory :




This program was designed in order to find the roots of an inserted polynomial equation and within a given domain , here are the steps that shows how the program works :

1. The Program asks the user to enter the desired equation.

2. The Program then checks the equation validity of format using the “Checker” function that tokenize the string , if it is correct it will proceed to the next step , else it will output the line “Wrong format” and exit the program .

3. The Program then determines the order of degree of the equation using the “order” function and outputs the value .

4. The Program then determines the powers and coefficients of the equation using the “power” function and outputs the coefficient in their order .

5. The function “power” then assign the coefficients after it’s been determined into the array .

6. The Program then will prompt the user to input the domain.

7. The program then using the “bisection” function will start finding the roots of the functions by adding the first value of the domain until it reaches the end of the domain

8. The program will then substitute the value in the function “f” , if the value of the function is = 0 , the roots will be printed.

## Program:

```
#include<iostream>
#include<math.h>
#include<cstring>
#include<string.h>
#include<stdio.h>
#define EP 0.000001
using namespace std;
void checker(char x[100]);
void order(char x[100], int *&y);
void power(char x[100], int *&y ,int *&o);
int f(double a, int q, int *&y, int *&o);
void bisection(double a, double b, char x[100], int *&y, int *&o, int q);
int main()
{
	char equation[100];
	int counter=1;
	cout << "please enter your equation \n";
	cin.getline(equation, 100, '\n');
	char equation1[100];
	strcpy(equation1, equation);
	checker(equation);
	for (int  i = 0; i < strlen(equation1); i++)
	{
		if (equation1[i] == '+' || equation1[i] == '-')
		{
			counter++;
		}
	}
	int *cooficients;
	cooficients = new int[counter] {};
	order(equation1, cooficients);
	int *up = new int[counter] {};
	power(equation1, cooficients, up);
	int a, b;
	cout << "please enter the domain " << endl;
	cin >> a >> b;
	bisection(a,b,equation1,cooficients,up,counter);
	delete[] cooficients;
	delete[] up;
	system("pause");
}
void bisection(double a, double b , char x[100], int *&y, int *&o , int q )
{
	do
	{
		if (f(a,q,y,o)==NULL)
		{
			cout << "The root : " << a << endl;
			a = a + 0.1;
		}
		a = a + EP;
	} while (a <= b);
	if (f(b,q,y,o)==NULL)
	{
		cout << "The root : " << b << endl;
	}
}
int f(double a, int q, int *&y, int *&o )
```

```
{
	double fx = 0;
	for (int e = 0; e < q; e++)
	{
		fx += y[e] * pow(a, o[e]);
	}
	if (fx > 0.00001 || fx < -0.00001)
	{
		fx = 1;
	}
	return fx;
}
void power(char x[100], int *&y, int *&o)
{
	int i = 1;
	int j = 0;
	int z = strlen(x);
	while (j < z)
	{
		if (x[j-1]=='^')
		{
			o[i-1] = o[i-1] * 10 + (x[j] - 48);
			i++;
		}
		else if ((x[j - 1] == '+' || x[j - 1] == '-') && (x[j + 1] == '+' || x[j + 1] == '-'))
		{
			if (x[j]=='x')
			{
				o[i-1] = 1;
				i++;
			}
			else
			{
				o[i-1] = 0;
				i++;
			}
		}
		else if ((x[j + 1] == '+' || x[j + 1] == '-')&&(x[j]=='x'))
		{
			o[i-1] = 1;
			i++;
		}
		else if (isdigit(x[j]) && (x[j + 1] == '+' || x[j + 1] == '-'))
		{
			o[i - 1] = 0;
			i++;
		}
		j++;
	}
}
void order(char x[100], int *&y)
{
	int i = 1;
	int j = 1;
	int o = 0;
	int z = strlen(x);
	if (x[0] == 'x')
	{
		y[0] = 1;
	}
	else if (x[0] == '-')
	{
```

```
        if (x[1] == 'x')
        {
                y[0] = -1;
        }
        else
        {
                y[0] = (-(y[0] * 10 + (x[1] - 48)));
                if (isdigit(x[2]))
                {
                        y[0] = (y[0] * 10 - (x[2] - 48));
                        j++;
                }
                j++;
        }
}
else
{
        y[0] = y[0] * 10 + (x[0] - 48);
        if (isdigit(x[1]))
        {
                y[0] = y[0] * 10 + (x[1] - 48);
        }
}
if(x[2]=='^')
{
        o =o*10+( x[3]-48);
}
else if (x[1] == '^')
{
        o = o * 10 + (x[2] - 48);
}
else if (isdigit(x[0])&&(x[1]!='x'))
{
        o = 0;
}
else
{
        o = 1;
}
while (j < z)
{
        if ((x[j + 1] == 'x' || x[j - 1] == '+'||x[j-1]=='-'))
        {
                if (x[j - 1] == '-'&& x[j+1]=='x')
                {
                        y[i] = (-(y[i] * 10 + (x[j] - 48)));
                }
                else if (x[j-1]=='+'&&x[j+1]=='+')
                {
                        i--;
                }
                else if (x[j] == '-'&& x[j+1]=='x')
                {
                        y[i] = -1;
                }
                else if ((x[j]=='+'&&x[j+1]=='x'))
                {
                        y[i] = 1;
                }
                else if (x[j - 1] == '-')
                {
                        if (x[j + 1] == '+' || x[j + 1] == '-')
```

```
                        {
                                y[i]= (-(y[i] * 10 + (x[j] - 48)));
                        }
                        else if (isdigit(x[j + 1]))
                        {
                                y[i]=(-(y[i] * 10 + (x[j] - 48)));
                                y[i]= (-(y[i] * 10 - (x[j] - 48)));
                        }
                        else
                        {
                                y[i] = (-(y[i] * 10 + (x[j] - 48)));
                        }
                    }
                    else
                    {
                            y[i] = y[i] * 10 + (x[j] - 48);
                            if (isdigit(x[j + 1]))
                            {
                                    y[i] = y[i] * 10 + (x[j+1] - 48);
                                    j++;
                            }
                    }
                if (x[j + 2] == '^')
                {
                       if(0*10+(x[j+3]-48)>o)
                       o =0*10+( x[j + 3]-48);
                }
                i++;
            }
                    j++;
        }
        cout << "the order of the equation = " << o << endl;
        for (int c = 0; c < i; c++)
        {
                cout << "cooficients number " << c+1 << " : " << y[c] << endl;
        }
}
void checker(char x[100])
{
        char* ptr;
        ptr = strtok(x, "+""-");
        while (ptr != NULL)
        {
                if (isdigit(ptr[0]))
                {
                        if (ptr[1] == 'x')
                        {
                                if (ptr[2] == '^')
                                {
                                        if (isdigit(ptr[3]))
                                        {
                                        }
                                        else
                                        {
                                                cout << "wrong format" << endl;
                                                getchar();
                                                exit(0);
                                        }
                                }
                                else if (ptr[2] == 0)
                                {
                                }
```

```
                        else
                        {
                                cout << "wrong format" << endl;
                                getchar();
                                exit(0);
                        }
                }
                else if (ptr[1] == 0)
                {
                }
                else if (isdigit(ptr[1]))
                {
                }
                else
                {
                        cout << "wrong format" << endl;
                        getchar();
                        exit(0);
                }
        }
        else if (ptr[0] == 'x')
        {
                if (ptr[1] == '^')
                {
                        if (isdigit(ptr[2]))
                        {
                        }
                        else
                        {
                                cout << "wrong format" << endl;
                                getchar();
                                exit(0);
                        }
                }
                else if(ptr[1]==0)
                {
                }
                else
                {
                        cout << "wrong format" << endl;
                        getchar();
                        exit(0);
                }
        }
        else
        {
                cout << "wrong format" << endl;
                getchar();
                exit(0);
        }
        ptr = strtok(NULL, "+""-""\0");
    }
}
```

# Examples:

1-

```
D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
2x^3-3x^2-3x+2
the order of the equation = 3
cooficients number 1 : 2
cooficients number 2 : -3
cooficients number 3 : -3
cooficients number 4 : 2
please enter the domain
-5
5
The root : -1
The root : 0.5
The root : 2
Press any key to continue . . .
```

2-

```
D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
5x^5+3x^2+x+1
the order of the equation = 5
cooficients number 1 : 5
cooficients number 2 : 3
cooficients number 3 : 1
cooficients number 4 : 1
please enter the domain
-2
2
The root : -0.86
Press any key to continue . . .
```

3-

```
D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
x^5+3x^4+5x-2
the order of the equation = 5
cooficients number 1 : 1
cooficients number 2 : 3
cooficients number 3 : 5
cooficients number 4 : -2
please enter the domain
-5
0
The root : -2.71
The root : -1.67
Press any key to continue . . .
```

4-

```
D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
2x^3-3x^2-3x+2
the order of the equation = 3
cooficients number 1 : 2
cooficients number 2 : -3
cooficients number 3 : -3
cooficients number 4 : 2
please enter the domain
1
3
The root : 2
Press any key to continue . . .
```

5-

```
D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
x^3+4x^2-9x-3
the order of the equation = 3
cooficients number 1 : 1
cooficients number 2 : 4
cooficients number 3 : -9
cooficients number 4 : -3
please enter the domain
-5
0
The root : -0.297
Press any key to continue . . .
```

6-

```
D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
5x^3+3x^2+x+2
the order of the equation = 3
cooficients number 1 : 5
cooficients number 2 : 3
cooficients number 3 : 1
cooficients number 4 : 2
please enter the domain
-5
5
The root : -0.88486
Press any key to continue . . .
```

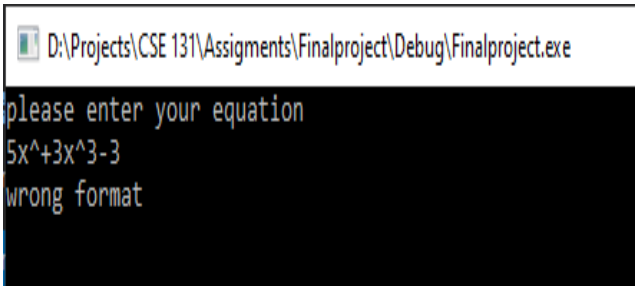
7-

```
D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
7x^4-7
the order of the equation = 4
cooficients number 1 : 7
cooficients number 2 : -7
please enter the domain
-5
5
The root : -1
The root : 1
Press any key to continue . . .
```

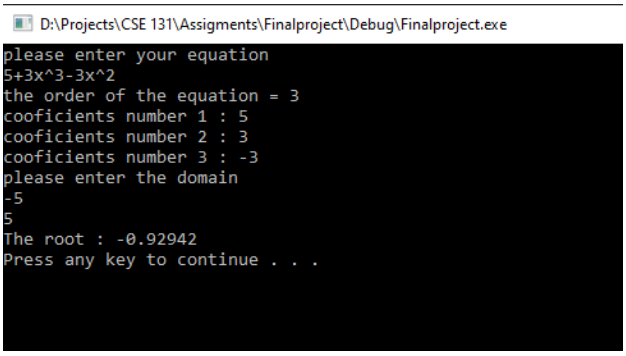
8-

```
D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
9x^2+4x^3-3
the order of the equation = 3
cooficients number 1 : 9
cooficients number 2 : 4
cooficients number 3 : -3
please enter the domain
-5
5
The root : -2.07597
The root : -0.69435
The root : 0.52031
Press any key to continue . . .
```

9-


D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
5x^+3x^3-3
wrong format

10-


D:\Projects\CSE 131\Assigments\Finalproject\Debug\Finalproject.exe
please enter your equation
5+3x^3-3x^2
the order of the equation = 3
cooficients number 1 : 5
cooficients number 2 : 3
cooficients number 3 : -3
please enter the domain
-5
5
The root : -0.92942
Press any key to continue . . .