

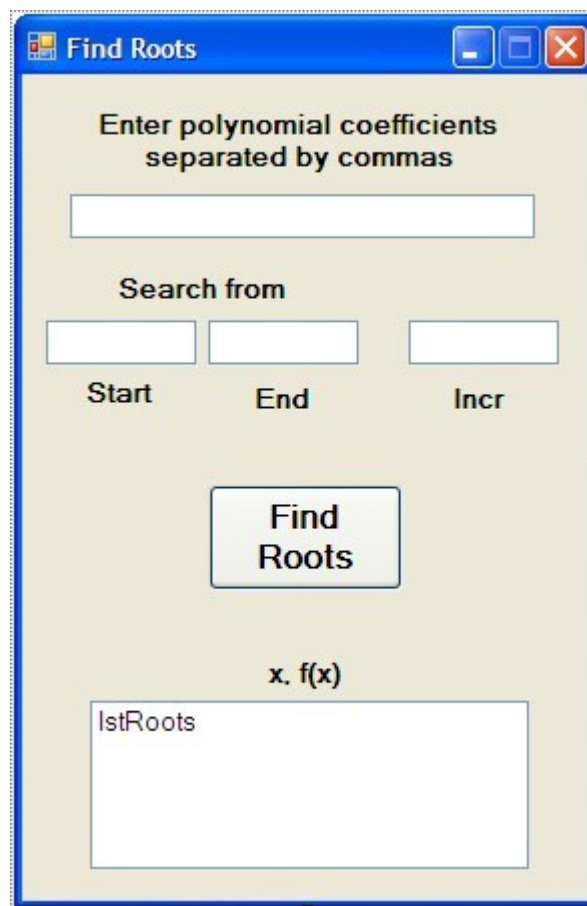
## CS 205 - Programming for the Sciences Spring 2008 - In-class Exercise for 01/22/08

Today's exercise is to start working on a program to compute the real roots of a polynomial function. For this exercise, you are given a C# project with a GUI already created. You are to write the C# code to provide the functionality.

Notes on finding the roots of a function are available on the course webpage.

Use a Web browser go to the course webpage <http://csserver.evansville.edu/~hwang/s08-courses/cs205.html>. Under today's date, right-click the link to the file FindRootsInClass.zip. Save this compressed folder to your network drive. From My Computer, browse to the compressed folder. Right-click on the compressed folder and select Extract All. Go through the steps of the Extraction Wizard ending with a window with the extracted solution folder. Double-click into the folder, then double-click on FindRootsInClass.sln (the Visual Studio solution file). This will launch Visual Studio with the solution loaded.

The GUI design for the program has been completed. To see it, right-click on Form1.cs in the Solution explorer window and select View Designer. It should look like the following:



The screenshot shows a Windows application window titled "Find Roots". The window has a blue title bar with standard minimize, maximize, and close buttons. The main content area has a light beige background and contains the following elements:

- A text label: "Enter polynomial coefficients separated by commas"
- A single-line text input field.
- A text label: "Search from"
- Three single-line text input fields arranged horizontally, labeled "Start", "End", and "Incr" below them.
- A rectangular button labeled "Find Roots".
- A text label: "x. f(x)"
- A multi-line text area labeled "lstRoots" at the top left.

To enter a polynomial of degree  $n$ , list all of the coefficients separated by commas starting with the  $x^n$  term, then the  $x^{n-1}$  term, and so forth to the constant term ( $x^0$ ). Note that the coefficient of a "missing" term is 0. For example, for polynomial  $x^3 - 2x^2 - 2x + 1$ , the coefficients would be entered as "1, -2, -2, 1" (without the quotation marks), while for polynomial  $x^4 + 1$ , the coefficients would be entered as "1, 0, 0, 0, 1".

Right-click on Form1.cs in the Solution explorer window and select View Code. The code for splitting up the string of coefficients and converting them into double values and storing them into an array is given. Also given is a method for evaluating the entered polynomial function at an arbitrary value for  $x$  using a for loop statement. This code will be explained in class.

### **Assignment**

(10 points) Working in pairs, complete this program by completing the button handler to find the roots of the polynomial function using the endpoints and increment given by the user

We will work through the design and implementation of the root finding algorithm during class. When you are finished, demonstrate your program to an instructor.