UMass Lowell Dept. of Computer Science

91.461 GUI Programming I

Fall 2013 Semester, Section 201

Prof. Jesse M. Heines

Site Home
Teaching
Research
Advising
Calendar
Projects

Course Home Information Class Notes **Assignments** Piazza Grades Resources Logout

Assignment No. 7

Creating a Graphing Calculator Using the HTML Canvas

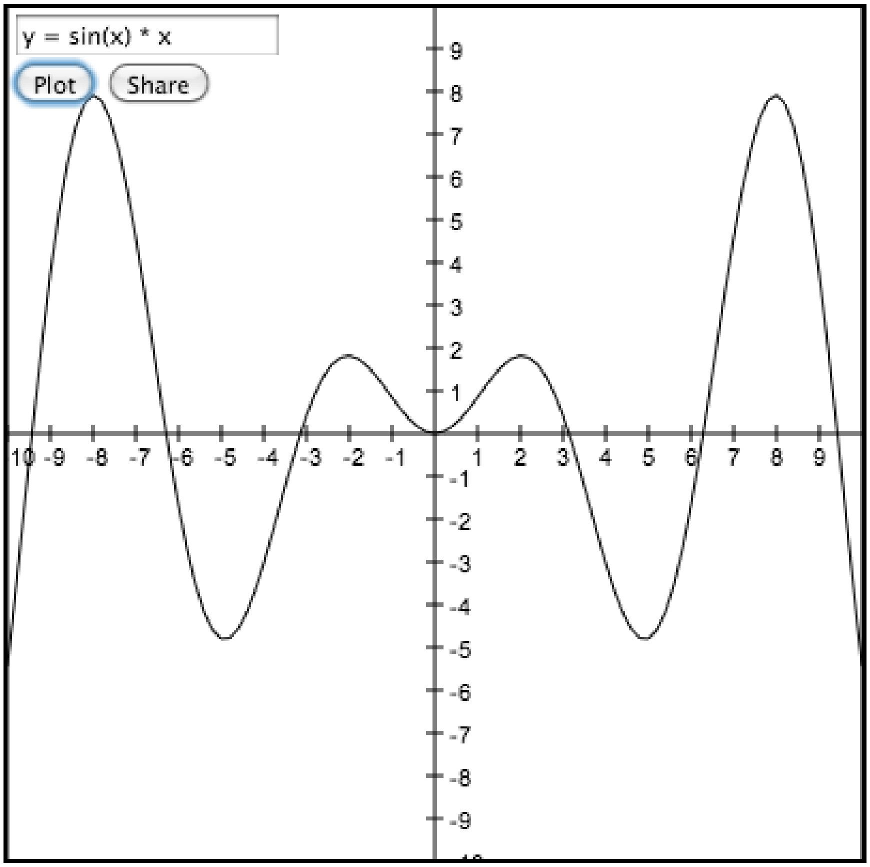
Date Due: **Thursday, November 21, 2013**

Contents Top

What This Assignment Is About Top

This assignment centers around using the HTML5 Canvas API to build a graphing calculator application. The core of the program will be from the in-class tutorial given by Curran, and your task is to “finish the application by adding more features and refining its appearance.

Here is an example of how the end result might look:



y = sin(x) * x

What You Are To Do Top

1. Review the content at https://github.com/curran/screencasts/tree/gh-pages/grapher. This GitHub repository

contains example code you can draw from to complete this assignment. Additional examples (including a working but sloppily coded graphing calculator) can be found at https://github.com/curran/HTML5Examples.

2. Starting with the core code created in class, the grapher application can be extended and improved in almost limitless ways. You may draw from any example code you like (just be sure to cite sources in comments). Code you wrote for previous assignments may also be useful. For this assignment, choose **any four** of the following features and implement them yourself by modifying and extending the code from the in-class tutorial.

    - add labeled X and Y axes (vertical and horizontal lines with "X" and "Y" text labels)
    - add tick marks (small lines perpendicular to axes that designate units)
    - add tick mark labels (text next to tick marks that shows their values, for example -1 0 1 2 3 ...)
    - define the arrangement (size and positioning) of UI elements using CSS
    - polish the aesthetic of the application appearance (for example borders, colors, and fonts) using CSS
    - add a user interface to modify the viewing window (minX, maxX, minY, maxY)
        - note: this is very similar to Assignment No. 5
    - add error handling (with feedback in the user interface) for invalid mathematical formulas
        - note: this is very similar to Assignment No. 6
        - see the mathJSExample for how to catch parse errors and extract the error message
        - suggested solution: when the expression is invalid, display the error message in red on the canvas in place of the plotted function
    - add a polar coordinates mode
    - add a parametric coordinates mode

3. Test your page thoroughly. Try to anticipate all the errors that a user might make, whether intentional or unintentional. Make sure that you handle each possibility.

4. Have a friend run your application and try to "break" it. Plug any "holes" that your friend finds in your application and/or its validation scheme.

Submitting Your Assignment for Grading

Top

As always, submit your assignment using the Assignment Submission Form. This form will lead you through providing all the information we need to evaluate your work, confirming that it is correct, and then e-mailing that information to our teaching assistant and me with a copy to yourself. Be sure to select **91.461 Assignment No. 7: Creating a Graphing Calculator Using the HTML Canvas** from the assignment drop-down list on that form.

Important Note: If you do not receive an e-mail in your CS account within a few minutes confirming that your assignment has been submitted, something has gone wrong. Try submitting again or contact me.

How You Will Be Graded

Top

This assignment will be graded on a 20-point system with points awarded as follows. Please note that the lists of features provided below are not meant to be exhaustive. They are merely representative of the types of things we are looking for in each grading category. As always, 20% of your grade is for documentation.

Criteria (numbers in parentheses are the points that can be earned for that bullet item)	***Possible Points***
Program Integrity / Design • (4) one of four features is implemented • (4) two of four features are implemented • (4) three of four features are implemented • (4) four of four features are implemented	16
Source Code Documentation and Formatting • user name and pertinent contact information appear in all source files • ***all*** files contain adequate explanatory documentation that is is meaningful and does not merely echo code • ***all*** files are properly indented and formatted with adequate white space for readability • ***any sources used are cited in comments embedded within code***	4

This is document https://teaching.cs.uml.edu/~heines/91.461/91.461-2013-14f/461-assn/CreatingAGraphingCalculator-v01.jsp. It was last modified on Thursday, November 7, 2013 at 10:18 PM.