Journal Report 8
10/28/19-11/7/19
Jialin Tso
Computer Systems Research Lab
Period 2, White

Daily Log

Monday October 28

Generated random floats and integers n times. Converted decimals to fractions.

Tuesday October 29

Graphed user-inputted equations and evaluated functions at certain x values. Researched sympy solve to find intersection of two equations

Thursday October 31

Implemented intersection function, but realized that solve could only give x coordinate of intersection of two equations, not y coordinate.

Wednesday November 6

Figured out y coordinate for intersection. Began writing function to find all zeros of a function. Only able to find one zero, not all, with intersection(funct, 0). Therefore, began researching scipy's fsolve.

Thursday November 7

Discovered fsolve can find all roots, but similar to graphing calculator, user needs to estimate location of each zero. While in the process of testing fsolve, inputted a trig function into graph() function and realized that program was unable to recognize sin(x) and others requring "math.". Focused instead on making sure program can graph all functions. As of now, was not able to resolve issues, so began researching tkinter for Python GUI that would display calculator buttons.

Timeline

Date	Goal	Met
Sept 30	Find regression equations from user-inputted lists	Linreg, Polyreg, and Expreg can generate equations and graphs.
Oct 7	Convert answers from expression to words and resolve problem of word numbers separated by dashes	No dashes needed between numbers. Many non-numerical answers can be displayed in words.
Oct 14	Convert all calculator-recognized function answers to words	Success!
Oct 28	Add more functionalities to program	Program can generate random numbers n times, convert most decimals to fractions, evaluate functions, graph many equations, find intersection of two functions, and find a zero of a function.
Nov 11	Begin creating GUI for calculator.	

Reflection

Although many examples online showed solve outputting the coordinate (x,y) pair as the answer, when I tried using solve and solveset, both only outputted the x coordinate. I then tried indexing the first value in the solve array (i.e. the x coordinate) and inputting one of the equations into evaluate, which worked. I also discovered that converting decimals to fractions was different from on the calculator than with the program. On the calculator Ans to Frac gives me the correct fraction, but with the program, I think it truncates up to 16 or so decimal places, giving incorrect fractions. While attempting to resolve the issue of graphing functions that require "math.", I ran into the message, "AttributeError: 'Symbol' object has no attribute 'sin'". Apparently, this means that the computer translates np.sin(a) to a.sin(). An interesting fact is that when I graphed sine on my "trial.py" program where I directly input the equation, it worked. Something else I discovered was that optimize is different than what I expected because I need to create a separate function called f(x), then call f into optimize in order for it to work. However, I will keep these issues in mind for the future and begin working on learning about Python GUIs next week since that is more important.

Example of new input and output of current program: Input: intersection left three point five times x plus ninety comma seven times x right

Expression output: intersection(3.5*x+90,7*x)

Answer: [25.7142857142857, 180.000000000000]

Output: left bracket twenty-five point seven one four two eight five seven one four two eight five seven comma one hundred and eighty right bracket

Input: fraction left three point one four plus three point one four right

Expression output: fract(3.14+3.14)

Answer: 7070651414971679/1125899906842624 (Wow!, why not just 157/25?)

Output: seven quadrillion, seventy trillion, six hundred and fifty-one billion, four hundred and fourteen million, nine hundred and seventy-one thousand, six hundred and seventy-nine divided by one quadrillion, one hundred and twenty-five trillion, eight hundred and ninety-nine billion, nine hundred and six million, eight hundred and forty-two thousand, six hundred and

twenty-four

Input: zeroes left x to the power of three divided by two plus nineteen times x to the power of two divided by ten minus forty one times x divided by ten minus eleven divided by two right

Expression output: zeroes(x**3/2+19*x**2/10-41*x/10-11/2)

Answer: [-5.00000000000000, 0]

Output: left bracket minus five comma zero right bracket

Input: graph left x to the power of three plus two times x minus four right

Expression output: graph(x**3+2*x-4)

Graph: