Polynomial Evaluation

In mathematics, a polynomial is an expression consisting of variables (also called indeterminates) and coefficients, that involves only the operations of addition, subtraction, multiplication, and non-negative integer exponentiation of variables [<https://en.wikipedia.org/wiki/Polynomial>]. Chapter 5 of the textbook describes three methods for evaluating a polynomial, that is, supplying a value for the variable to produce a result: 1) standard; 2) Horner’s Method; and 3) Preprocessed coefficients. In this assignment you will implement methods 2 and 3.

# Assignment

Evaluate the following polynomial for the values [-20..20] stepping by 0.2. That is, evaluate *p(x)* for -20, -19.8, -19.6, …, 20.

The x-y scatter plot graph of the function (created with Excel) is as shown

Write functions to do so using two methods:

1. Horner’s Method
2. Preprocessed Coefficients
   1. You will have to perform the factoring operations to get the derived polynomials
   2. Make sure you reduce it to the simplest polynomials using the recurrence

Write the results to a file that can be read into a spread sheet program (Excel, Numbers, OpenOffice, etc.) and plotted. The file contents may look as follows:

-20.0, -1.022764437E9

-19.8, -9.508382467734915E8

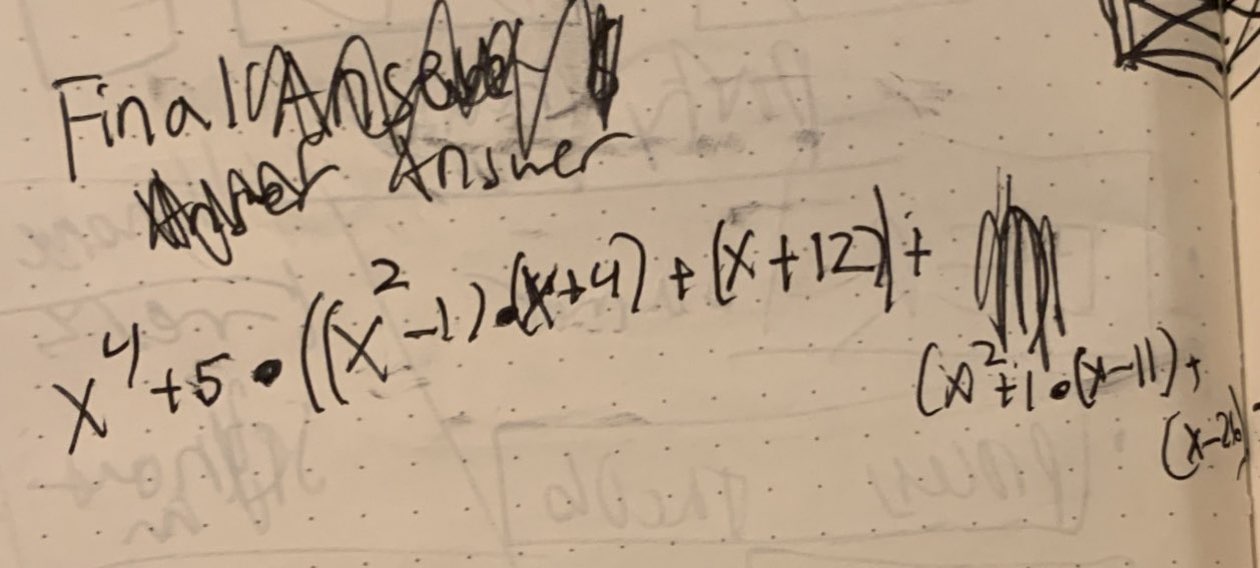
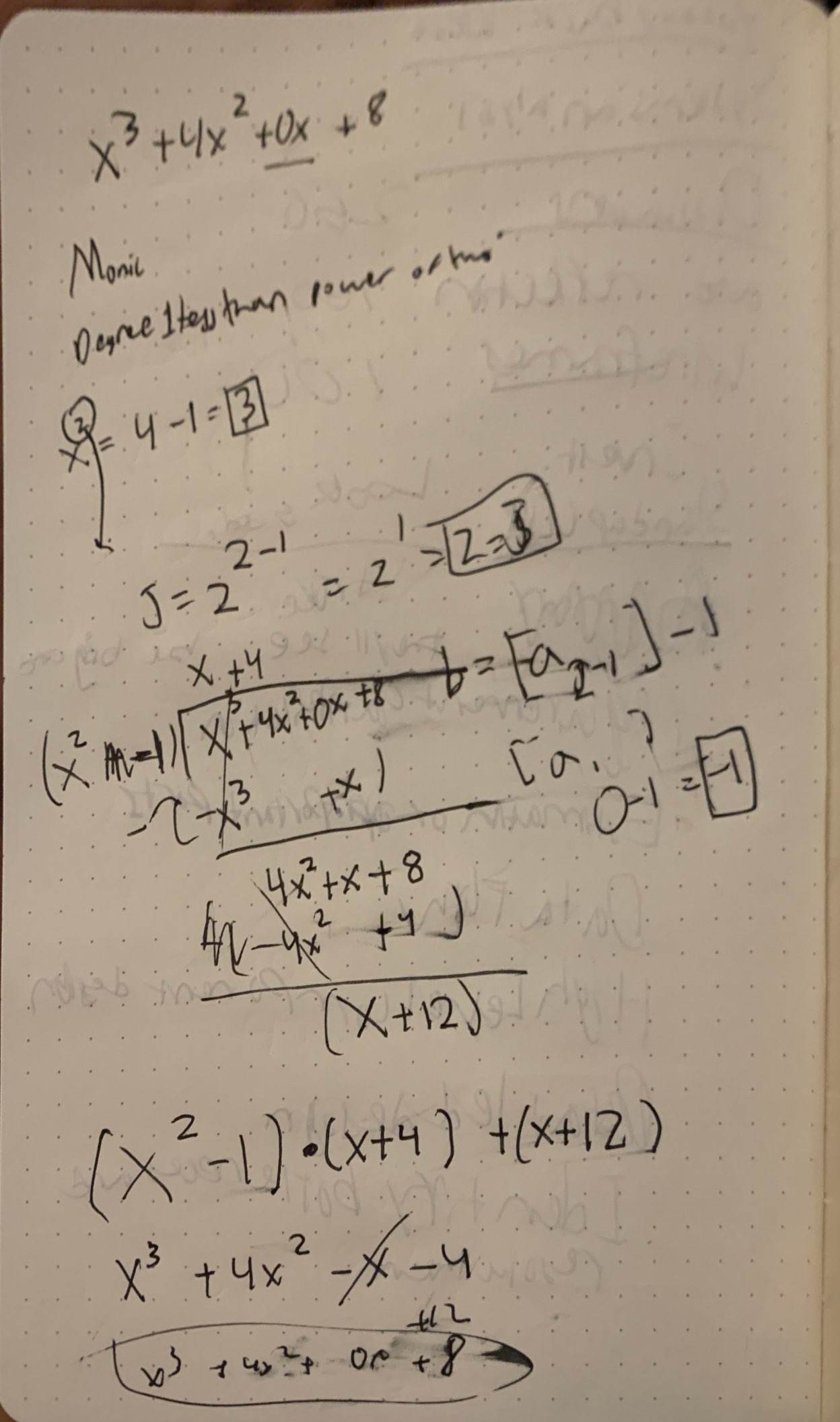
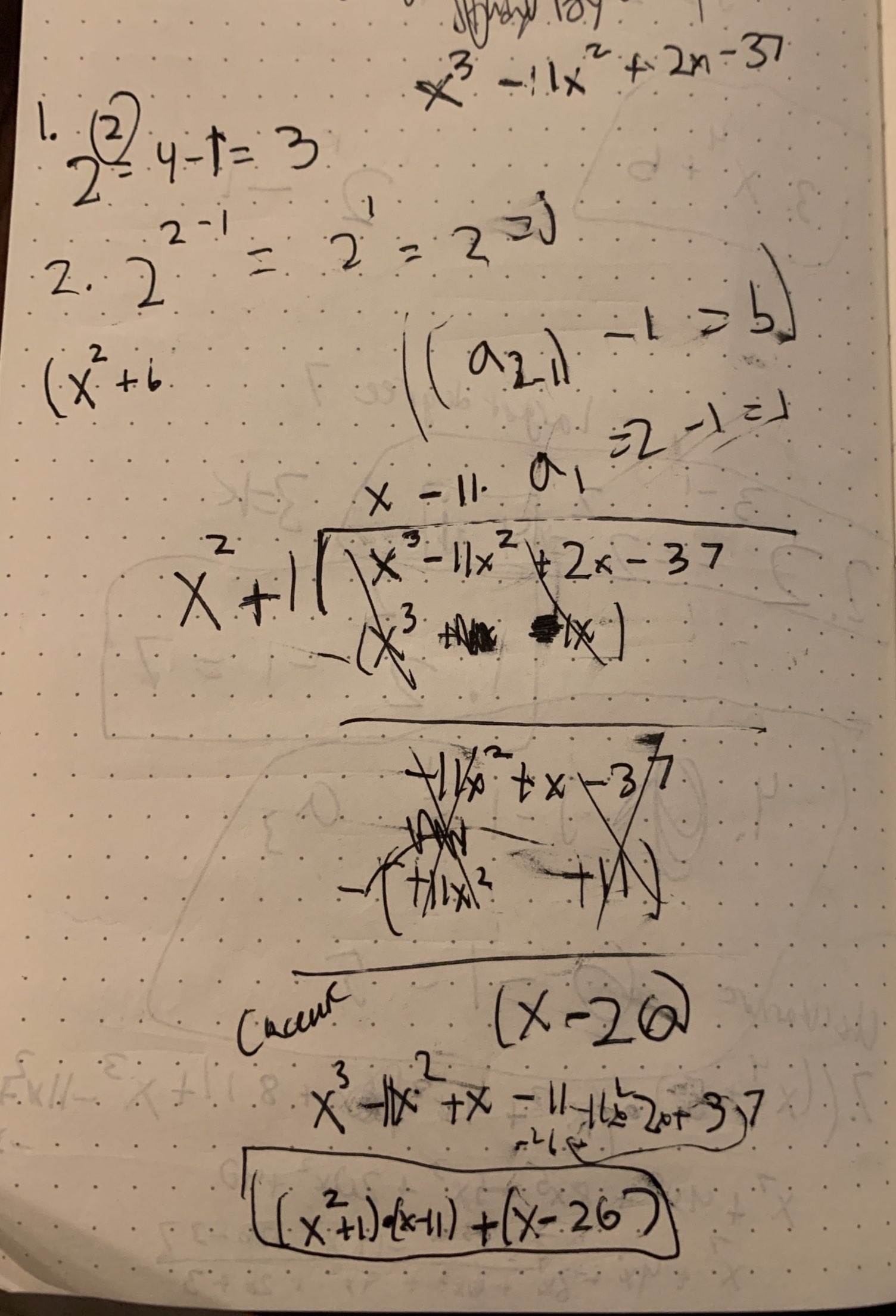
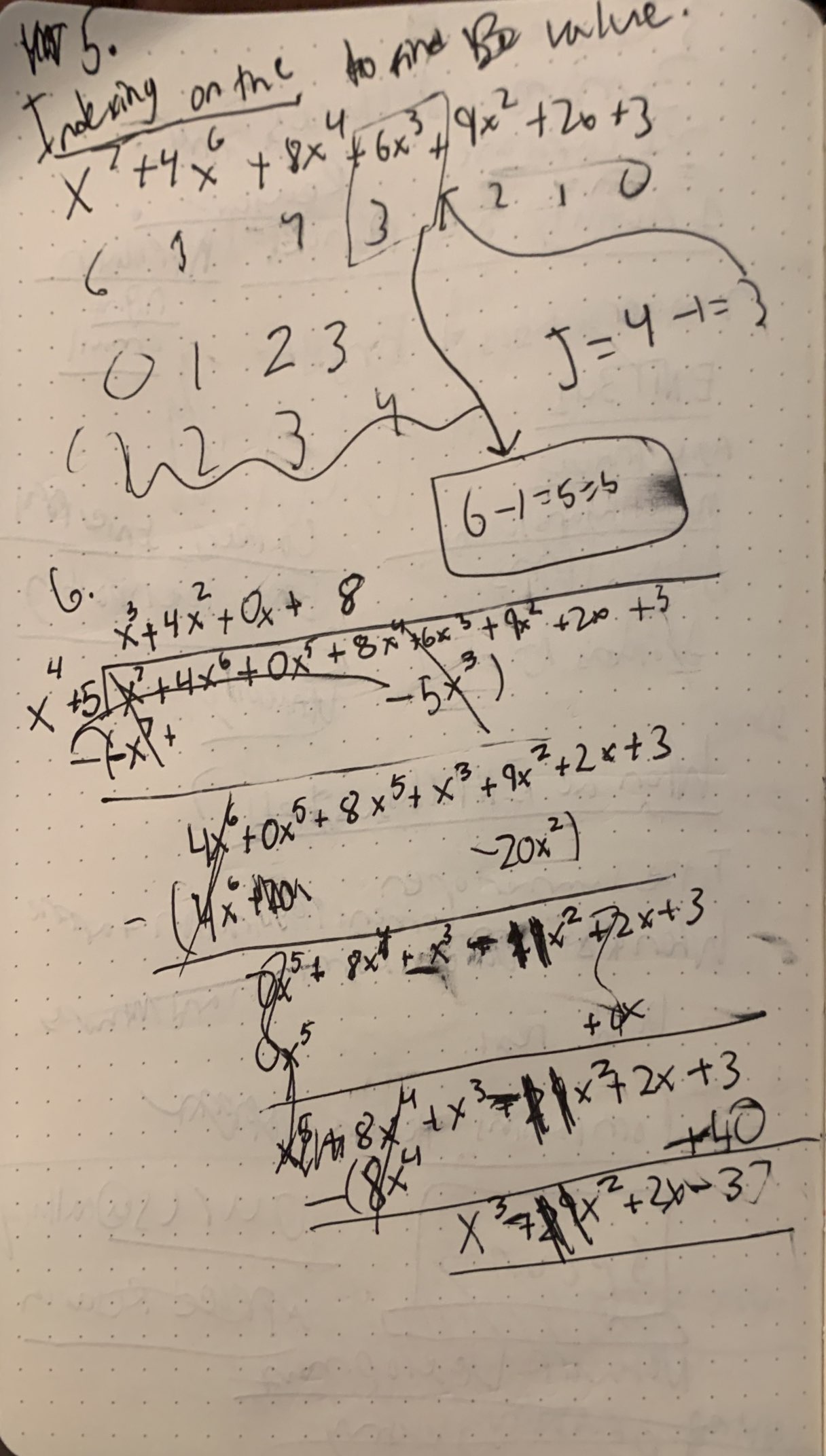
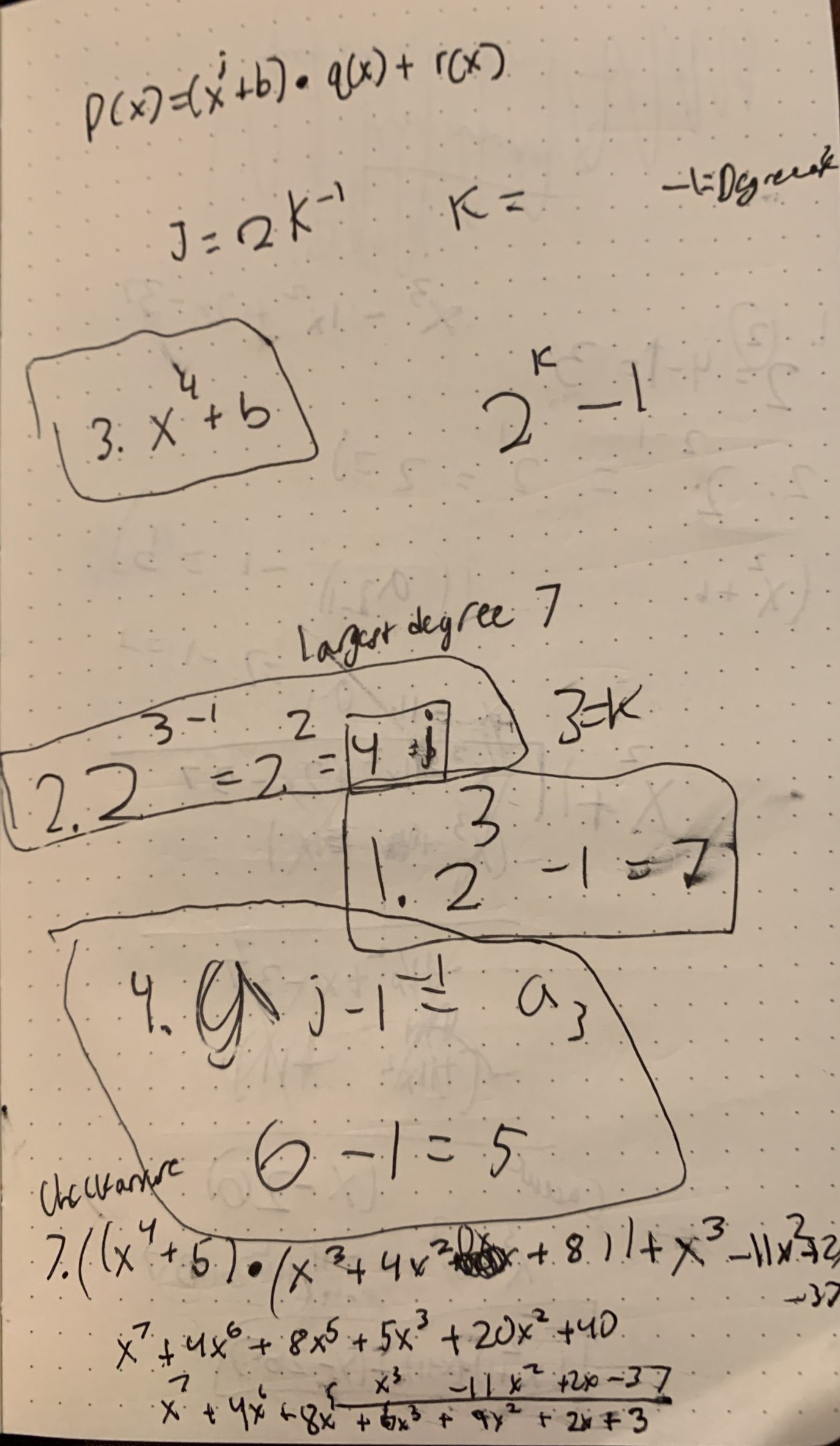
-19.6, -8.832861577858181E8

…

# Deliverables

* Source code
* A reflection document including
  + Plots of *x* vs. *p(x)* for each implementation
  + Essay describing successes and difficulties

Polynomial Evaluation Reflection

Factoring work for Preprocessed Coefficients 

Plots:

Reflection Essay:

This assignment took way longer than I needed it too. I started by inaccurately assuming that the equation give was the same in the book. That quick assumption resulted in about three hours of absolute pain and suffering. I kept tirelessly looking over the same tens lines of trying to figure out why the equation was close but not completely accurate. I thought the fact that that doubles have that extra 0.00000002 on them. I tirelessly changed instantiation values in vain only to find I had made no progress. After meeting with you I found I was using the wrong equation. After applying the new equation and doing the division, checking each preprocessed coefficient by doing the evaluation after I did the equation. After plugging the new equation and changing out the hard coded equation for Horner’s method I managed to get it correct. So I have successfully calculating the above equation using Horner’s and the preprocessed coefficient method. I managed to compare the answers with my graphing calculator to confirm my answer.