Polynomial Assignment Discussion

This program is to be run by using either a compiler or the command line to run the “DriverProgram.class” file, which will access the “Polynomial.class” to test all of it's functions. The program will grab a polynomial from “PolynomialFile.txt” the user is expected to then enter a second polynomial (Specifically the second polynomial asked for in the subject outline, 7a^9 + 2a^8 + 5a^6 + 2a^5 + 2a^3 + 9a^2 – 7a). This is done by first entering the size of the polynomial, then entering each monomial in a coefficient then exponent order.

The program will then handle the rest and display the results.

I'd also like to point out that despite the subject outline calling for three constructors (Empty, File and Scanner), I chose to instead implement these as functions, as this is better programming practice and a post on the forums explicitly allowed this so long as both a File reader and Scanner reader were implemented. As I noted above, implementing the polynomial builders as functions is a better practice and means that existing polynomials can still be built upon.

Overall I had little difficulties with this assignment, with most of the difficulties arising from simple programming mistakes, such as referencing the wrong variable in a for loop, which may be a result of either already established programming knowledge or as a result of the slightly more than average knowledge I have of mathematics.

The two functions I had the most difficulty with were the polynomial addition and Scanner reading functions. I had difficulty with the addition function due to the possibilities that the polynomials could be empty or of different lengths. In the end I merely decided to add an effectively empty monomial into the start of every polynomial. This way every polynomial could be guaranteed to always have at least one monomial, equivalent to 0a^0.

I also had difficulties in implementing the Scanner reading function, as every time I tried to implement it using a pre check loop where the user had to enter a specific term to end the inserting of monomials, wouldn't work, simply endlessly repeating. Rather than potentially waste time on the task I simply cheated my way around the task and required the size of the polynomial to be inserted first, which gave the program a hard number to count up to in iterations of entering coefficients and exponents.

Lastly I chose not to implement differing operators or removing exponents or variables in the polynomial, as these would have simply been by and large time sinks. It is a simple enough check using case statements or if-then-else statements to check if a monomial is negative, or the last, or an exponent of 0 or 1, and change the output to match, but ultimately I decided against this, as it is the exact same mathematically.

Scott Gardner

11489878