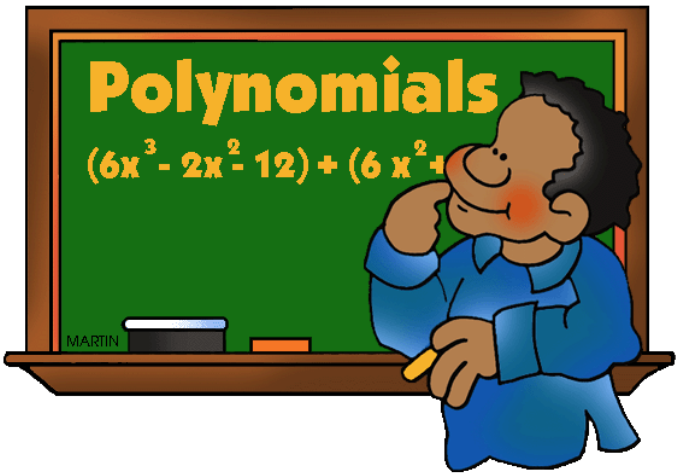
UNIVERSITY OF PUERTO RICO
MAYAGUEZ CAMPUS
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING



Polynomial Solver Language Proposal

By:
Eduardo Vázquez
Rosedanny Ortiz
Daniel Navarro
Carmelo González

Dr. Wilson Rivera
Department of Electrical & Computer Engineering
University of Puerto at Mayaguez
ICOM4036 – Programing Languages
September 15, 2015

Introduction

Polynomial Solver Language (PSL) is a brand new programming language design to solve the most common mathematical operations on all types of polynomials. The main motivation is based under the impression that writing polynomials in any current language is a little hard, because the user needs to implements data structure and it will not be able to use any type of variable. Therefore using PSL the user will be able to express any polynomial expression with any variable he desires and perform any of the mathematical functions, with the main goal of focusing on the math part and keep practicing of solving polynomials. It would be like solving exercises on paper but using a programming language.

Language Features

Some of the basic features the language will provide are, but not limited to:

- Assignment of a function to an independent variable, so it's easier to perform arithmetic operations with others expressions.
- Simple assignment of values to all variables in the expression, or across multiple polynomials that shares them, and evaluating them with given values.
- The ability to solve basic arithmetic operations such as addition, subtraction, multiplication and division with others polynomials.
- Finding the derivative of simple polynomial functions.
- The display of the given polynomial, and all resulting polynomials from any operation, in their standard form for easier understanding.
- Simple implementation for finding the summation of a series of polynomial equation given the interval.

Example of the Program

The user should be able to specify which type of variable his polynomial will have:

- var x (means polynomial will be of x)
- var y (means polynomial will be of y)

The number that appears before a given variable means is the constant, the number that appears after the given variable is the exponent and numbers that are alone before or after a math operand is a constant. For example:

- 3x2 + 2x3 + 8x1 + 9 is the representation of ($3x^2 + 2x^3 + 8x + 9$)

User will be able to perform some of math operations like:

- deri (3x2 + 2x3 + 8x1 + 9 = 0) which means the derivative of ($3x^2 + 2x^3 + 8x + 9$)
- the output of that expression is 6x1 + 6x2 + 8

Software Requirements

1. Applications written on this language must be able to run by command line.
2. The application created will run on the windows environment.
3. The scanner and parser will be implemented using the python or java programming language.
4. The user's machine should have at least 1GB of RAM available and a Dual Core processor clocking at 2.0 GHz or better.
5. The user should have the necessary Python or Java packages already installed before developing on PSL.

Specifications

1. Programs will be written using the Unicode character set and will end each statement using a line terminator.
2. Features pre-established names for each corresponding entity.
3. Support pages which can be decorated with annotations like any other declaration.
4. The sequence of execution of the program is controlled by each corresponding statement.
5. Variables must have an assigned value when access to its values occurs.

Project Plan Timeline:

Phase	Stages	Dates
Phase 2: Language Translator	• Lexical Analyzer • Syntax Analyzer • Intermediate Code	• September 21 to October 9, 2015 • October 12-30, 2015 • November 2-13, 2015 Due date: November 15, 2015
Phase 3: Final Report and Demo	• Language Tutorial • Reference Manual • Translator Architecture • Descriptions: interfaces, software development environment, test methodology.	• November 16-20, 2015 • November 23-27, 2015 • November 30 to Final Exam Date Due date: Final Exam Date

**Dates are subject to change.*