DOCUMENTATION

POLYNOMIAL CALCULATOR

CONTENTS

1. Assignment Objective

(i) main objective of the assignment: Designing and implementing a polynomial calculator that allowes the user to insert 2 polynomials and perform some operations
(ii) the sub-objectives in a table/list:

- *To implement a main Polynomial class that contains the split of the coefficients*
- *To implement each operation in different classes*
- *To use Swing to create an interface that the user can utilise*

The sub-objectives represent the steps required to follow for achieving the main objective. Each sub-objective will be described and the section in which it will be addressed should be mentioned.

2. Problem Analysis, Modeling, Scenarios, Use Cases

A. Problem Analysis

A polynomial is a mathematical expression that consists of an addition of terms called monomials. Polynomial operations (addition, subtraction, multiplication, division, derivation, integration) are fundamental to mathematics, engineering and physics.

B. Modeling and scenarios

The user can add 2 polynomials in the 2 textbox on the interface. After that, they can chose one of the buttons that represent a specific operation of polynomials. The result will appear at the bottom of the page.

The operations that can be performed are: addition, subtraction, multiplication, derivation, integration

The user will write whatever 2 polynomials and will perform some sort of operation on them regarding the size and the types of monomials.

C. Use cases

The execution steps are as following:

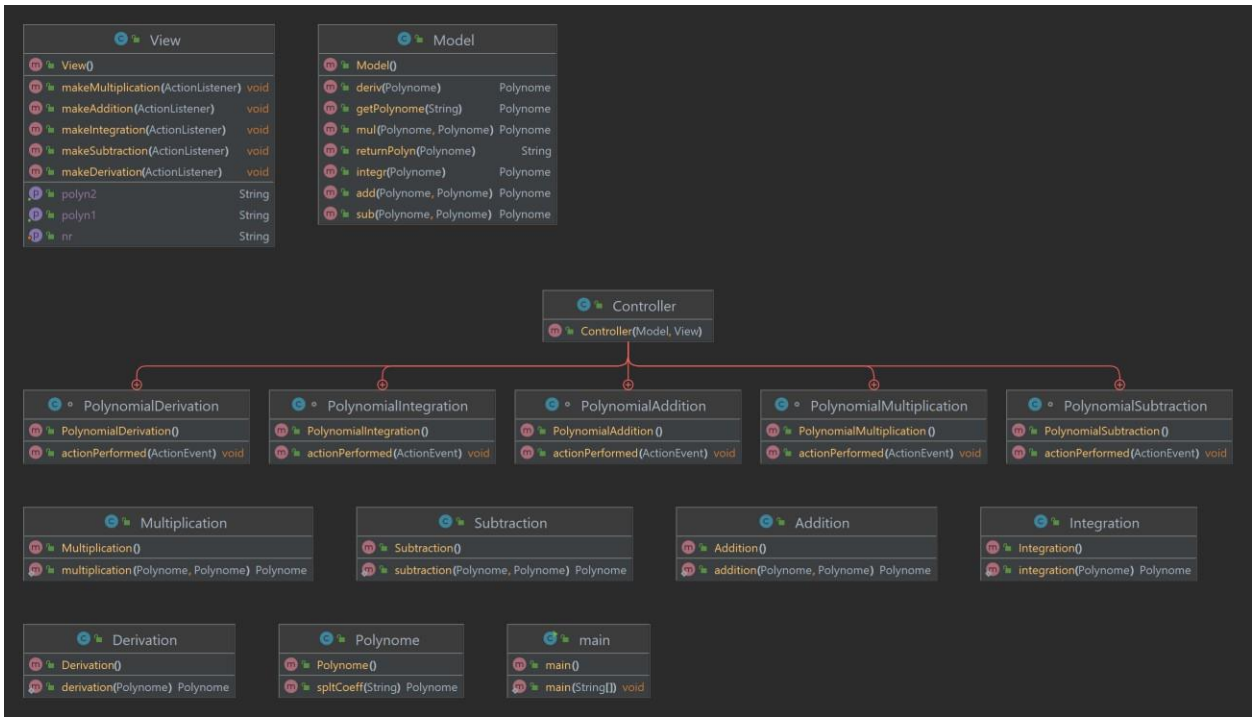
In the text boxes the 2 polynomials will be written.

After that, one of the 5 buttons representing the operations will be pressed by the user and the specific operation will be executed

In the end the result will appear on the screen, at the bottom of the page More

than 1 operation can be executed on a set of poynomials

3. Design



Data Structures
I used prymitive data types to store data and variables that I need. The coefficients and powers of the polynomials are stored in a HashMap that also has an ArrayList fore storing more coefficients having the same power

Packages
The method used for the interface is the MVC(Model View Controller), that allowes a much faster and organised way of creating an user interface.
-Model: It directly manages the data, logic and rules of the application
-View: is just a visual representation of a model, and does not handle user input. It contains the buttons and text fields that can be seen on the interface
-Controller: Accepts input and converts it to commands for the model or view

4. Implementation

My project has a total of 10 classes.

The Polynome class

- It contains the split of the monomials and the storing in a HashMap that will be used to perform the operations
- This is included in one method *splitCoeff();*

The operation classes *Addition, Subtraction, Multiplication, Derivation, Integration)

- All of this classes contain a single method, the one that performs the specific operation by computing coefficients and storing the result into HashMaps that will later be printed as Strings

The View class

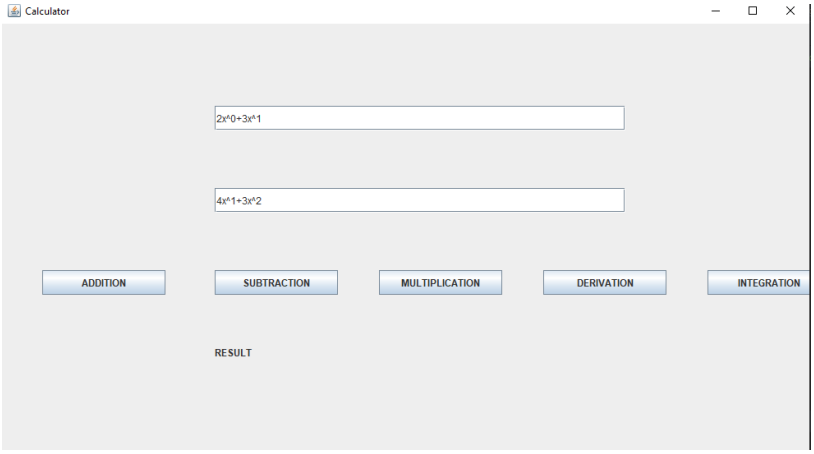
- Contains all the buttons and text fields that are used for the user interface and also the initialization of the action that will be performed by each of them

The Model class

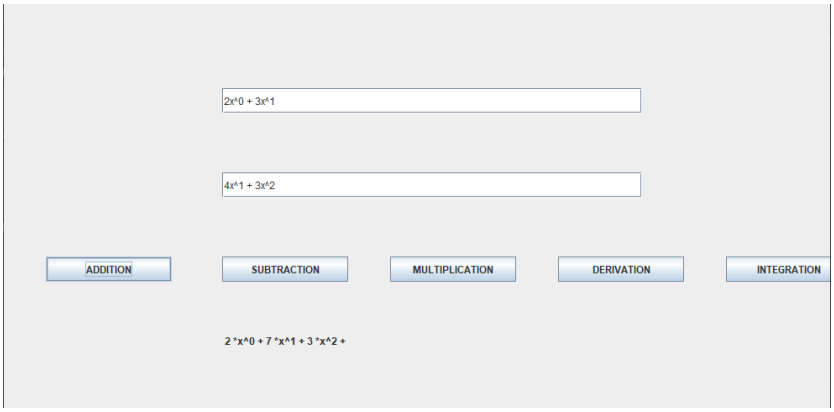
- Contains methods of each operation that will be performed by the calculator The Controller class
- Has 1 constructor used for the Model and View
- Contains the actions performed by the buttons

5. Results

Introducing the polynomials

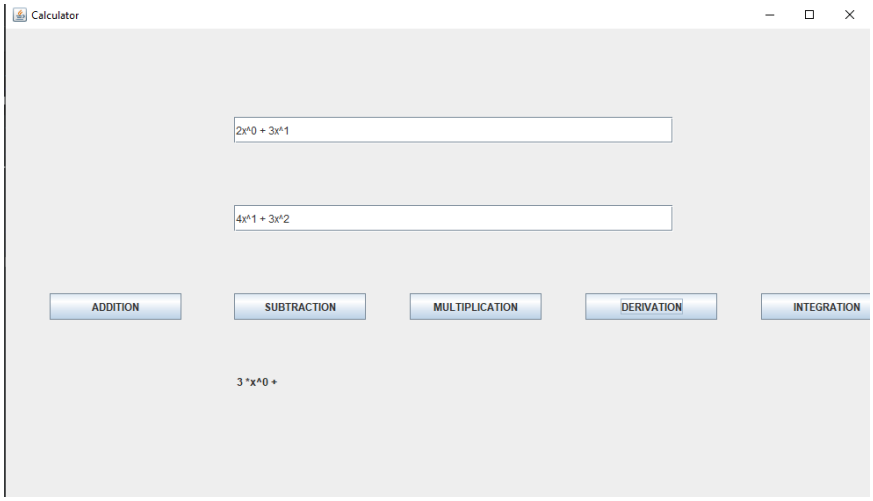


Addition operation

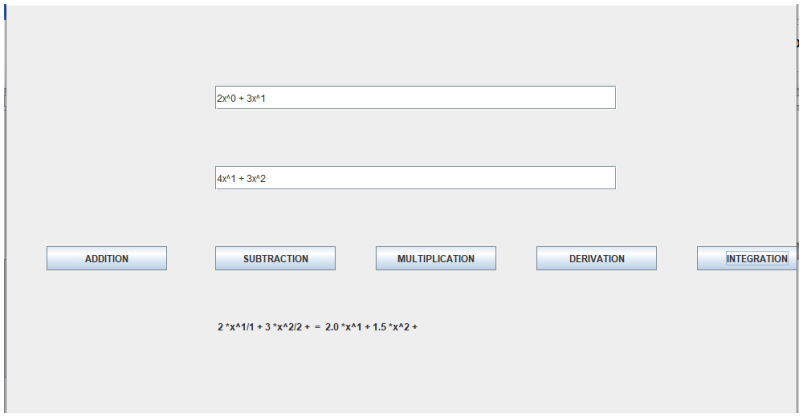


Subtraction operation

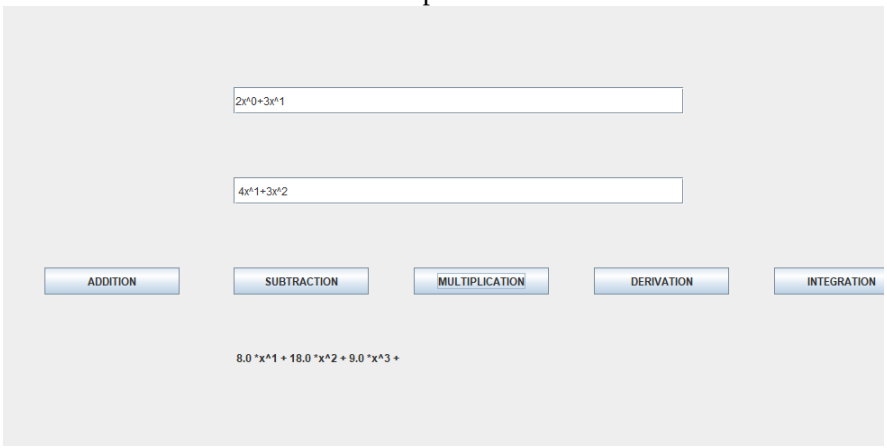
Derivation



Integration



Multiplication



6. Conclusions

This project helped me to understand more about MVC and also the ways of organizing the code in a good way.
For the future I would like to implement the division operation.