## TEHNICAL UNIVERSITY OF

CLUJ-NAPOCA

POLYNOMIAL PROCESSING

Name: Calugar Marius Adrian

Group:30422

1.Objective :

Propose ,design and implement a system for polynomial processing .Consider the polynomials of one variable and integer coefficients.

The main objective of this task is to have implemented and properly working the main operations that can be evaluated on the polynomials given such as: addition , subtraction ,division , multiplication , integration ,differentiation and so on .

To have a functional interface that can interact with the user when the users need to implement the operations on certain kind of polynomials .

Polynomial processing refers to the fact of taking as input from the user one or more polynomials (in this case two) and giving back as an output the result of this two polynomials after they were processed in some way .

A Monom is a data structure that holds a coefficient as a double data type and a degree as an integer data type .

We represented the polynomial given as input as a List on Monoms and then we took this list and split it into particular elements that were further divided to take from them the coefficient and the degree needed.

2.Problem analisys:

In the first steps of application the parameters of the interface will be set and this will be displayed to the user .

The user will introduce the polynomials that will be processed in the following natural form ,for example :”2x^3+3x^2+5x+1” or “-x^3+3x=7”.

After both the polynomials are introduced in the designated JTextField area , the user will have the possibility to choose between many operations that can be preformed on these polynomials .

The first operation is the addition that starts from the polynomial with the highest degree and then iterates the second polynomial while comparing the monom’s degree , if the degree is the same another Monom is created with this degree and the sum of the coeficients of Monoms found .

This operation acts on both the polynomials and outputs a totally new polynomial so it is stored in the Processes Class.

The second operation is the subtraction of polynomials that is again in the Class Processes that takes two polynomials as input and make the addition of the first with the negative of the second polynomial .

Then the multiplication that takes both the polynomials given as input and multiplies them term by term and after that is simplifies the polynomial that needs to be displayed .

The division of the polynomials takes both of them into consideration and starts with the polynomial that has the highest degree , after the process is done it gives as an output the quotient and reminder polynomial .

The differentiation process in the Polynomial Class because it takes as input only the first polynomial given , and it differentiates it and then gives the result as output .

The integration process is also in the Polynomial Class and takes only the first polynomial ,integrates in term by term and then outputs the new created polynomial.

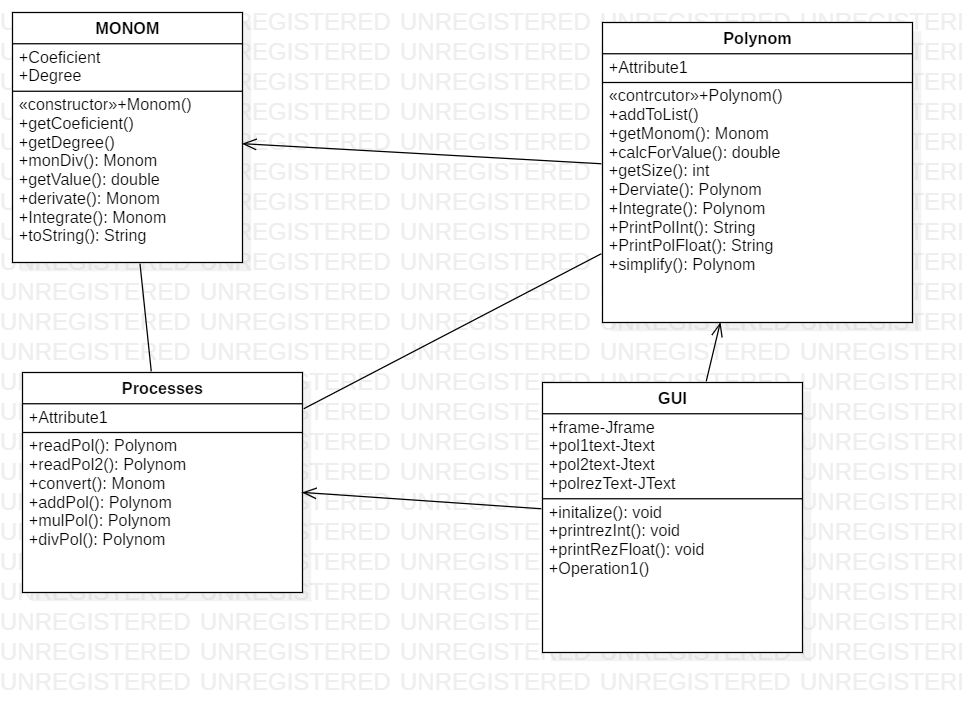
The secondary objectives of the project are to get used to processing and parsing strings given as input and putting them in well defined structures created by the designer of the application while processing them in a direct manner.

The designer also needs to get used to using the GUI interface that the developing program provides so that the user interface can be created and programed in a way that is easy to be understood by the users that will interact with the application .

We made the method of parsing the string given as input as flexible as possible so that the user can enter the polynomial in a natural way , similar to the way one would write it as a normal mathematical expression .

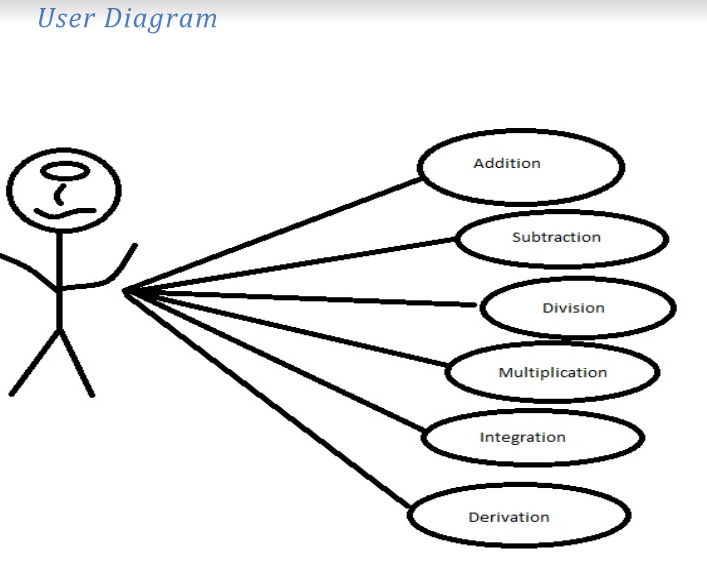
This task is wisely chosen as a starting point in the process of creating java applications because it uses the main concepts of Object Oriented Programming , subject which we studied last semester.

3 Designing :



I decided that the best way to implement this task to divide it into smaller parts so I represented the polynom given as a list of Monoms and a part of the operations are on those monoms so in this way we can simplify the processes needed to complete the task .

As we can see in the UML diagram the Class Polynom contains many elements of the type Monom so that we can better preform the operations needed .



The interface was implemented in the GridLayout style because this provided the necessary functionality to better view the user interface .

We divided the interface into 3 main parts as follows, the first part divided into 2 columns where the user introduces the input data in the form of polynomials .

The second area is reserved for the buttons that operate on the input polynomials then this process is displayed on the third part of the interface.

I did not design any verifications for the input of the polynomials because I assume that the user will introduce the data in a normal and basic manner.

This part may be considered as a future development of the project , where a user could only display a certain type of polynomial otherwise the program will give an error and ask the user to introduce another polynomial structure as input .

4.Implementation :

The first Class that I implemented is the Monom class which has as attributes a coefficient as a double data type and a degree as in integer data type .

The constructor in this class takes as input two parameters , a double that will take place of the coefficient and an integer that will take place of the degree.

The method getCoefficient return the coefficient of the current Monom , this method will be used a lot in the processes that involve the input polynomials.

The method getDegree is of int type and returns the degree of the current monom.

The static method div takes as parameters two Monoms and creates a new Monom that has as coefficient the division of the coefficient’s of the two Monom’s given as parameters .The new Monom has as degree the substraction of the degrees of the two Monoms given as parameters.

The method getValue takes an integer as parameters and returns the value of that Monom in that certain value given as parameter .The result is returned as a double data type.

The method Derivate takes as a parameter a Monom and returns a new Monom that represents the mathematical differentiation of the Monom given as parameter. This method works on a general case no matter the degree or the coefficient of the Monom given as input.

The method Integrate takes as a parameter a Monom and gives as an output a new Monom that represents the integration of the Monom given as parameter .The new Monom won’t have a c constant because this will over-complicate the processes needed for our application to work .

Now we will describe the use of the Class Polynom in the design of the Polynomial Processing application .

The class Polynom has a single attribute that is a ArrayList of Monoms that will hold the polynom that is given from the user input as a string then it will be converted based on a regular expression in order to fit the necessary form for it to be placed into a monom .

The constructor of this Class is just a basic constructor that initializes the list of Monoms .

The method addToList takes a Monom as a parameter and introduces the monom in the current polynom at the end , this method will be used very often in our design.

The method getMonom , takes an integer as an input and this integer is seen as an index.

It return the Monom from the specified index.

The method calcForValue returns a double data type that represents the value of the polynomial expression at the particular point given as input parameter.

The simple method getSize just returns the length of the current polynomial.

The static method Derivate takes as a parameter a Polynom data type and creates a new Polynom that represents the mathematical expression of the differentiation of the polynom given as parameter .

The static methos Integrate takes a single parameter that is of the type Polynom and creates a new Polynom .That new polynom takes the values of the integration of each monom that forms the polynom given as an input parameter.

The method printPolInt takes a polynom data type as input and gives a string as output.This string represents the practical view of the polynomial given as an input parameter.This method only takes in consideration integer data type coefficients .

The method printPolFloat takes a polynom data type as a parameter and gives a string as an output , this string will be placed on the final JTextField where the output polynomial is represented. This method only handels float coefficient data types .This method will be used only in the case of integration where the coefficient of the output polynomial will be of double type .

The method mulConst will take a Polynom and an integer as an input and it will give as output a new Polynom .This method will multiply each coefficient of the polynom with the constant given in the parameter list .

This method will be used in the case of subtraction where the output will be the addition between the first polynom and the second polynom multiplied with the constant -1 .

The method simplify will take a polynom as an input and il will write it as a simplified mathematical expression in this way we will be able to display it in a more natural way .

Next we will describe the functionality of the of the Class Processes .

The main operation of the application will be here ,for example the addition method will take 2 polynomials as input and it will create a new polynom that will represent the sum of the 2 polynomials given as input .

The method mulPol takes as input two polynomials and creates a new Polynom that represents the multiplication of the two polynomials given as input .This method works in a natural way where the polynomials are multiplied term by term to create a new polynom .

Conclusions:

In conclusion , the design of this project helped me understand the way the GUI framework works .

Some things I learn are the way of implementing the multiplication of two polynomials in a efficient way .

The addition could be made on a different logic and maybe this will make it be more efficient on a later development .

The integration technique used in this application works exactly as a normal mathematical integration but without the c constant because this will need more complex calculations that are not necessary here .

I learned to implement certain operations that represent processes on data types and classes created by myself.

The main thing that I learned from the development of this project is the basic design and operations and structures of the Object Oriented Programming .

The future development of this application are many .

For example we could implement operations on more than 2 polynomials in this way this application could be used in bigger projects designs .

We could introduce polynomials of many variables that could hold more complex operations such as partial differentiation .

We could also implement many operations in a row on the same pair of polynoms in this way we could have for example the addition of two polynomials and with the result we could divide it with another polynomial and so on .

A different further development could be to introduce polynomials of double type coefficient in this way we could create more complex models of operations .With these operations we could create new applications that involve these basic operations .