MINISTRY OF EDUCATION AND RESEARCH

TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA, ROMANIA

FUNDAMENTAL PROGRAMMING TECHNIQUES

ASSIGNMENT 1

POLYNOMIAL CALCULATOR

1. Requirements

Design and implement a polynomial calculator with a dedicated graphical interface through which the user can enter polynomials, select the operation to be performed (i.e. addition, subtraction, multiplication, division, derivative, integration) and display the result.

Note: *Consider the polynomials of one variable and integer coefficients.*

1.2 Design considerations

- Use an object-oriented programming design
  - Use encapsulation
  - Define classes such as Polynomial and Monomial
- Use the Model View Controller as an architectural pattern for designing the polynomial calculator.

1.3 Implementation considerations

- Use the Java programming language
- Use Java Swing for implementing the graphical user interface
- Use Regex for verifying the validity of the polynomials
- Use lists instead of arrays
- Use *foreach* instead of *for(int i=0…)*
- Implement classes with maximum 300 lines of code (except for the UI classes)
- Implement methods with maximum 30 lines of code
- Use the Java naming conventions (https://google.github.io/styleguide/javaguide.html)

1.4 Testing considerations

- Use JUNIT for testing the application.

2. Deliverables

- A **solution description document** (minimum 2000 words, Times New Roman, 10pt, Single Spacing) organized according to the structure specified in the **Lab Description** document.
- **Source files** – will be uploaded on the personal ***bitbucket*** account following the steps (check the **Lab Resources**, and **Lab Description** documents for more details):
  - Create a repository on ***bitbucket*** named according to the following template *PT2020_Group_ LastName_FirstName_Assignment_1*
  - Push the source code and the documentation (**push the code not an archive with the code**)
- Share the repository with the user **utcn_dsrl,** or send an invitation to the address **utcn.dsrl@gmail.com**

3. Evaluation

The assignment will be graded as follows:

Requirement	Grading
Minimum to pass • Classes with maximum 300 lines (except the UI classes) • Methods with maximum 30 lines • Java naming conventions • Object-oriented programming design (encapsulation, *Polynomial* and *Monomial* classes) • Lists instead of arrays • *foreach* instead of *for(int i=0…)* • Graphical user interface • Implementation of the addition and subtraction operation • Documentation	5 points
Implementation of the multiplication operation	0.5 points
Implementation of the division operation	1 point
Implementation of the derivative operation	0.5 points
Implementation of the integration operation	0.5 points
Design according to the Model View Controller architectural pattern	1 points
Regular expressions and pattern matching for extracting the polynomial coefficients	0.5 points
Junit for testing	1

4. Bibliography

- Junit:
  - http://www.mkyong.com/tutorials/junit-tutorials/
- Add libraries Eclipse:
  - http://www.wikihow.com/Add-JARs-to-Project-Build-Paths-in-Eclipse-%28Java%29
- Swing:
  - http://zetcode.com/tutorials/javaswingtutorial/
  - http://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html
- Books:
  - SCJP, Java 8 & JavaFX
  - http://ptgmedia.pearsoncmg.com/images/9780321927767/samplepages/0321927761.pdf