DOCUMENTATIE

TEMA 1

NUME STUDENT: VLASE ELENA-RAFAELLA

GRUPA: 30225

# CUPRINS

[1. Obiectivul temei 3](#_Toc95297885)

[2. Analiza problemei, modelare, scenarii, cazuri de utilizare 3](#_Toc95297886)

[3. Proiectare 4](#_Toc95297887)

[4. Implementare 6](#_Toc95297888)

[5. Rezultate 9](#_Toc95297889)

[6. Concluzii 9](#_Toc95297890)

[7. Bibliografie 9](#_Toc95297891)

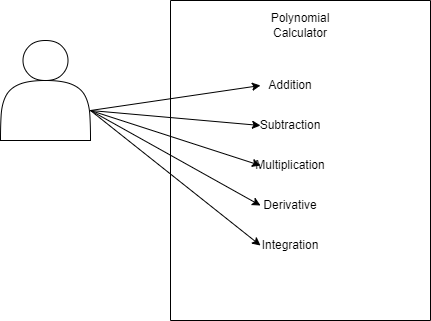
# Obiectivul temei

Obiectivul primei teme este implementarea unui calculator polinomial cu următoarele operații: adunare, scădere, înmulțire, împărțire a două polinoame, derivare și integrare a unui polinom.

Pentru a atinge acest obiectiv a fost nevoie de implementarea următoarelor:

* Interfață grafică (GUI)
* Structura de date Polinom folosind TreeMap
* Metode de convertire din String in Polinom, folosind regex
* Operațiile de adunare, scădere, înmulțire, derivare și integrare
* Funcții auxiliare precum toString, formatare Double etc.

# Analiza problemei, modelare, scenarii, cazuri de utilizare



Analiza:

Polinomul este o expresie de forma



Accentul se pune pe implementarea corectă a operațiilor pe polinoame. Am plecat de la premisa că gradele monoamelor din polinoame sunt valori întregi, iar coeficienții pot avea și valori reale. După efectuarea oricărei operații, termenii din polinom ce au coeficientul 0 nu vor fi afișați.

Modelare:

Structura de date “Polynomial” conține un TreeMap de monoame, key este gradul și value este coeficientul.

Scenarii:

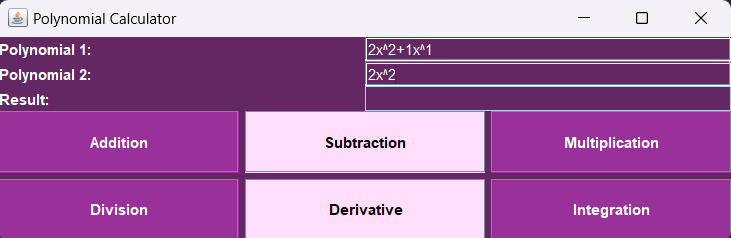
Pentru funcționare:

- Necunoscuta se notează cu “x”

- Termenii sunt de forma ax^b

- Nu se pun spații

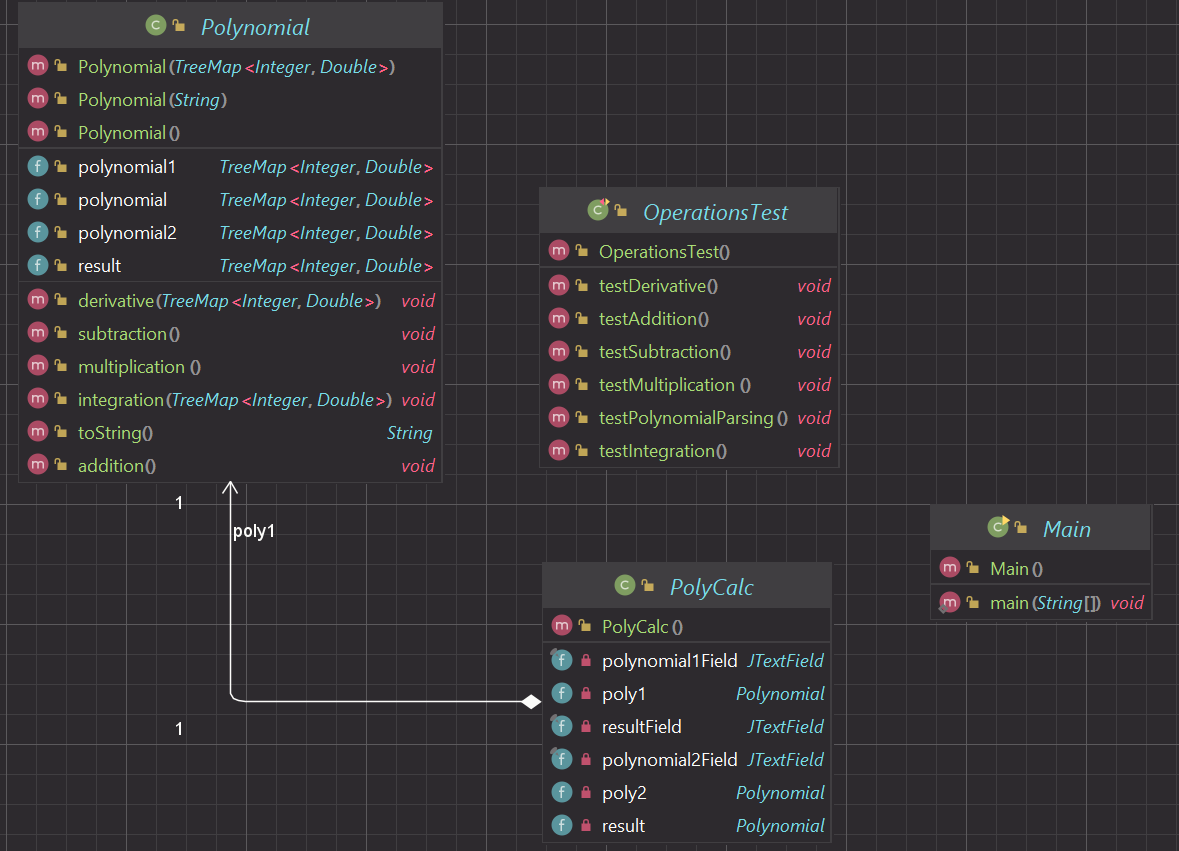
Use cases:



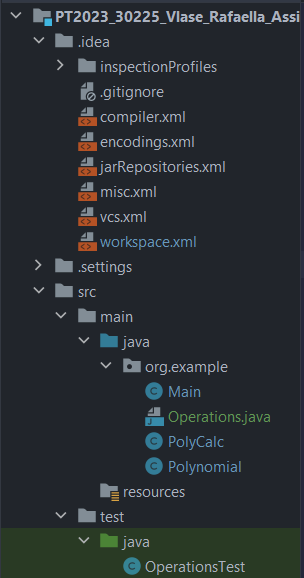
Interfața grafică este formată din două câmpuri de text pentru a insera polinoamele, un câmp unde va fi afișat rezultatul și șase butoane corespunzătoare fiecărei operații: adunare, scădere, înmulțire, împărțire, derivare, integrare. La apăsarea fiecărui buton va apărea în câmpul “Result” rezultatul operației alese prin apăsarea unui buton.

# Proiectare

Diagrama UML:



Ierarhia de fișiere a proiectului:



PolyCalc este clasa în care am implementat interfața grafică, iar în Main se apelează constructorul acesteia pentru instanțierea interfeței grafice.

# Implementare

Clasa Polynomial este folosită pentru a realiza operații pe polinoame. Aceasta are următoarele atribute:

TreeMap<Integer, Double> polynomial: Un polinom reprezentat prin perechi de forma (grad, coeficient) stocate într-un TreeMap. Acesta este utilizat doar în constructorul Polynomial(TreeMap<Integer, Double> polynomial).

TreeMap<Integer, Double> polynomial1: Un polinom reprezentat prin perechi de forma (grad, coeficient) stocate într-un TreeMap. Acest polinom este folosit în operațiile de adunare, scădere și în derivare și integrare.

TreeMap<Integer, Double> polynomial2: Un polinom reprezentat prin perechi de forma (grad, coeficient) stocate într-un TreeMap. Acest polinom este folosit în operația de înmulțire.

TreeMap<Integer, Double> result: Rezultatul unei operații pe polinoame, reprezentat prin perechi de forma (grad, coeficient) stocate într-un TreeMap.

Această clasă are următoarele metode:

public Polynomial(): Constructor fără parametri care inițializează polinoamele 1, 2 și rezultatul cu un TreeMap gol.

public Polynomial(TreeMap<Integer, Double> polynomial): Constructor cu un parametru de tip TreeMap<Integer, Double> care inițializează polinomul cu polinomul dat ca parametru.

public Polynomial(String polynomialString): Constructor cu un parametru de tip String care inițializează polinomul 1, 2 și rezultatul cu valorile date în formatul String-ului dat ca parametru.

public void addition(): Metoda care calculează suma polinoamelor 1 și 2 și salvează rezultatul în polinomul rezultat.

public void subtraction(): Metoda care calculează diferența polinoamelor 1 și 2 și salvează rezultatul în polinomul rezultat.

public void multiplication(): Metoda care calculează produsul polinoamelor 1 și 2 și salvează rezultatul în polinomul rezultat.

public void derivative(TreeMap<Integer, Double> polynomial1): Metoda care calculează derivata polinomului 1 și salvează rezultatul în polinomul rezultat.

public void integration(TreeMap<Integer, Double> polynomial1): Metoda care calculează integrala polinomului 1 și salvează rezultatul în polinomul rezultat.

@Override public String toString(): Metoda care convertește polinomul rezultat într-un String cu formatul standard al polinoamelor.

Clasa PolyCalc reprezintă un calculator de polinoame cu o interfață grafică utilizator (GUI). Clasa conține câteva câmpuri și metode care permit utilizatorilor să introducă două polinoame și să efectueze diferite operații, precum adunarea, scăderea, înmulțirea, derivarea și integrarea lor.

Câmpurile clasei includ:

"polynomial1Field": Un câmp de text care permite utilizatorilor să introducă primul polinom.

"polynomial2Field": Un câmp de text care permite utilizatorilor să introducă al doilea polinom.

"resultField": Un câmp de text care afișează rezultatul operației efectuate de utilizator.

"poly1": O instanță a clasei Polynomial care reprezintă primul polinom.

"poly2": O instanță a clasei Polynomial care reprezintă al doilea polinom.

"result": O instanță a clasei Polynomial care reprezintă rezultatul unei operații între două polinoame.

Metodele clasei includ:

"PolyCalc()": Constructorul clasei care inițializează interfața grafică și câmpurile necesare pentru a efectua operațiile.

"addButton.addActionListener()": O metodă care adaugă un ascultător la butonul de adunare. Când butonul este apăsat, această metodă crează două instanțe ale clasei Polynomial, le adaugă împreună și afișează rezultatul în "resultField".

"subButton.addActionListener()": O metodă care adaugă un ascultător la butonul de scădere. Când butonul este apăsat, această metodă crează două instanțe ale clasei Polynomial, le scade și afișează rezultatul în "resultField".

"mulButton.addActionListener()": O metodă care adaugă un ascultător la butonul de înmulțire. Când butonul este apăsat, această metodă crează două instanțe ale clasei Polynomial, le înmulțește și afișează rezultatul în "resultField".

"derButton.addActionListener()": O metodă care adaugă un ascultător la butonul de derivare. Când butonul este apăsat, această metodă creează o instanță a clasei Polynomial și calculează derivata polinomului, apoi afișează rezultatul în "resultField".

"intButton.addActionListener()": O metodă care adaugă un ascultător la butonul de integrare. Când butonul este apăsat, această metodă creează o instanță a clasei Polynomial și calculează integrala polinomului, apoi afișează rezultatul în "resultField".

# Rezultate

# Concluzii

Cu ajutorul acestei teme am învățat să lucrez cu git și tot acum am folosit pentru prima data Maven.

O posibilă dezvoltare ulterioară a calculatorului de polinoame este implementarea metodei de împărțire sau implementarea unei funcționalități ce permite afișarea unui istoric al rezultatelor operațiilor efectuate anterior.

# Bibliografie

<https://www.geeksforgeeks.org/java-swing-jpanel-with-examples/>

<https://docs.oracle.com/javase/tutorial/uiswing/index.html>

<https://stackoverflow.com/questions/36490757/regex-for-polynomial-expression>

<https://topic.alibabacloud.com/a/use-a-regular-expression-regex-to-match-polynomials-polynomial-regexpolynomial_1_31_32668196.html>

<https://dsrl.eu/courses/pt/>