

DOCUMENTATIE

TEMA 1

NUME STUDENT: SORIN TURDA
GRUPA:30222.....

CUPRINS

1.	Obiectivul temei.....	3
2.	Analiza problemei, modelare, scenarii, cazuri de utilizare	3
3.	Proiectare	4
4.	Implementare	4
5.	Rezultate	4
6.	Concluzii.....	5
7.	Bibliografie	5

1. Obiectivul temei

Obiectiv principal: Implementarea unui calculator de polinoame cu interfata grafica unde utilizatorul poate insera polinoamele caror li se vor efectua operatiile de: adunare, scadere, inmultire, derivare, impartire si integrare

Obiectiv secundar:

- Analiza problemei si realizarea cerintelor
- Proiectarea calculatorului de polinoame
- Implementarea calculatorului
- Testarea calculatorului

2. Analiza problemei, modelare, scenarii, cazuri de utilizare

Cerinte functionale:

- Calculatorul de polinoame trebuie sa permita utilizatorului sa insereze polinoamele
- Calculatorul de polinoame trebuie sa permita utilizatorului sa selecteze operatia dorita pentru polinoame
- Calculatorul de polinoame trebuie sa efectueze operatiile corect

Cerinte non-functionale:

- Calculatorul trebuie sa fie intuitiv si usor de folosit de catre utilizator

Cazuri de utilizare: adunarea/scaderea/inmultirea/impartirea a doua polinome

Actor: utilizator

Scenariu de succes:

- Utilizatorul introduce 2 polinoame de forma $a_n x^n + a_{n-1} x^{n-1} + \dots + a_0$
- Utilizatorul apasa unul din butoanele add/subtract/divide/multiply
- Calculatorul efectueaza operatia selectata si afiseaza rezultatul

Scenariu alternativ:

- POLINOAMELE SUNT INTRODUSE INCORECT
- Aplicatia nu o sa faca nimic

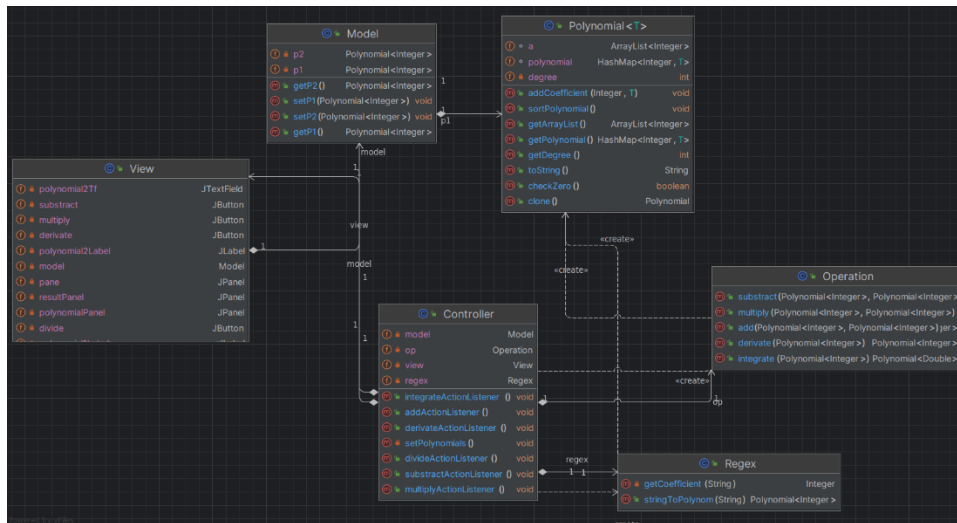
Derivarea/integrarea unui polinom

- Utilizatorul introduce primul polinom
 - Utilizatorul apasa unul din butoanele integrate/derivate

- Calculatorul derivatea/integreaza polinomul

Coeficientii polinoamelor sunt numere intregi.

3. Proiectare



4. Implementare

Clasa *Polynomial* este o clasa generica care foloseste un generic *T* de tip *Number* cu urmatoarele atribute:

- *ArrayList<Integer>* folosit pentru afisarea polinomului in ordinea descrescatoare a gradului
- *HashMap<Integer, T>* folosit pentru salvarea polinomului. Key-ul retine puterea coeficientului
- *Degree* retine gradul polinomului

Metode importante:

- *addCoefficient* adauga in *HashMap* monom cu monom

Clasa *Operation* are metodele care se ocupa de operatiile facute pe polinoame

Clasa *Regex* care transforma *String*-ul introdus de catre utilizator in polinom

5. Rezultate

Testare JUnit:

Adunare:

Polinim1: $+1x^3+3x^2-8x^1-10x^0$

Polinim2: $-5x^2+2x^0$

Rezultat: $+1x^3-2x^2-8x^1-8x^0$

Scadere:

Polinom1: $+1263x^{123}-128x^{23}-12x^2+1x^0$

Polinom2: $+29x^{123}+10x^{23}+1x^{12}+1x^{11}$

Rezultat: $+1234x^{123}-138x^{23}-1x^{12}-1x^{11}-12x^2+1x^0$

Derivare:

Polinom: $-78x^{99}+32x^{87}+11x^{75}-92x^{63}+54x^{51}-6x^{39}+84x^{27}-23x^{15}+98x^3+41x^0$

Rezultat: $-7722x^{98}+2784x^{86}+825x^{74}-5796x^{62}+2754x^{50}-234x^{38}+2268x^{26}-345x^{14}+294x^2$

Integrare:

Polinom: $+7x^9-2x^7+5x^5-8x^4+1x^3+4x^2-3x^1+6x^0$

Rezultat: $+0.7x^{10}-0.25x^8+0.83x^6-1.6x^5+0.25x^4+1.33x^3-1.5x^2+6x^1$

Inmultire:
Polinom1: $+1x^2+2x^1+1x^0$
Polinom2: $+1x^1+1x^0$
Rezultat: $+1x^3+3x^2+3x^1+1x^0$

6. Concluzii

Dezvoltari ulterioare:

- O implementare mai buna a clasei Polynomial in care se pot adauga chiar si valori reale, astfel operatiile pot fi facute pe orice tip de date
- Adaugarea unor exceptii si mesaje de eroare daca utilizatorul nu introduce corect polinoamele
- Operatia de impartie
- Interfata grafica mai prietenoasa, colorata si altfel aranjata

7. Bibliografie

1. <https://www.w3schools.com/>
2. <https://docs.oracle.com/javase/8/docs/api/>