DOCUMENTATIE

**Tema 1 – Calculator polinomial**

**NUME STUDENT: SCUTURICI VLAD LUCIAN**

**GRUPA: 30225**

# CUPRINS

[1. Obiectivul temei 3](file:///C:\Users\User\Desktop\l3\PT2023_Template_Documentatie_RO.doc#_Toc95297885)

[2. Analiza problemei, modelare, scenarii, cazuri de utilizare 3](file:///C:\Users\User\Desktop\l3\PT2023_Template_Documentatie_RO.doc#_Toc95297886)

[3. Proiectare 4](file:///C:\Users\User\Desktop\l3\PT2023_Template_Documentatie_RO.doc#_Toc95297887)

[4. Implementare 6](file:///C:\Users\User\Desktop\l3\PT2023_Template_Documentatie_RO.doc#_Toc95297888)

[5. Rezultate 8](file:///C:\Users\User\Desktop\l3\PT2023_Template_Documentatie_RO.doc#_Toc95297889)

[6. Concluzii 9](file:///C:\Users\User\Desktop\l3\PT2023_Template_Documentatie_RO.doc#_Toc95297890)

[7. Bibliografie 9](file:///C:\Users\User\Desktop\l3\PT2023_Template_Documentatie_RO.doc#_Toc95297891)

Obiectivul Temei

Obiectivul temei este implementarea unui calculator polinomial care va putea efectua operatiile de adunare, scadere, inmultire, impartire, derivare si integrare prin intermediul unei interfete grafice usor de utilizat, si totodata a functiilor matematice necesare.

Interfata grafica va permite utilizatorului sa introduca cele doua polinoame si apoi sa selecteze operatia, dupa care la apasarea butonului de calcul, rezultatul va fi afisat mai jos.

Analiza problemei, modelare, scenarii, cazuri de utilizare

Pentru a crea o aplicatie pentru calcularea polinoamelor, trebuie sa stabilim ce trebuie sa faca aplicatia si cum trebuie sa arate. Aplicatia trebuie sa permita utilizatorilor sa introduca polinoame cu coeficienti si sa efectueze operatiile matematice adunare, scadere, inmultire, impartire, derivare si integrare asupra acestora.

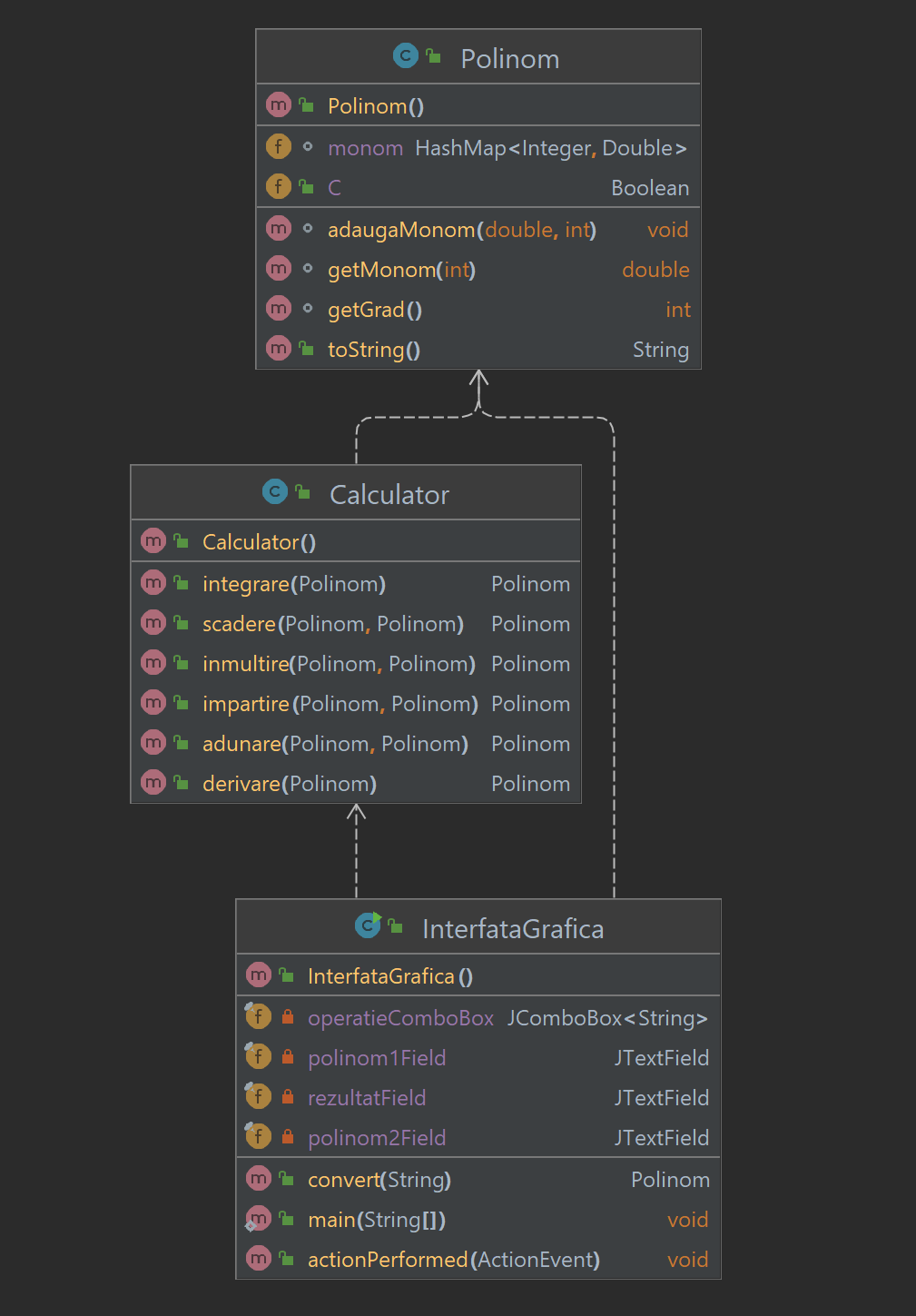
Aplicatia trebuie sa fie stabila si eficienta, iar timpul de raspuns cat mai mic, iar interfata grafica simpla si intuitiva, care sa ofere o experienta de utilizare fara probleme.

Scenarii:

Initial, utilizatorul are 2 casute in care va putea introduce 2 polinoame. Dupa introducerea polinoamelor, inainte de a apasa butonul “Calculeaza”, acesta va putea alege intre urmatoarele operatii:

1. Adunare: Rezultatul adunarii va fi afisat in casuta.
2. Scadere: Rezultatul scaderii primului cu al doilea polinom va fi afisat in casuta.
3. Inmultire: Rezultatul inmultirii va fi afisat in casuta.
4. Impartire: Rezultatul impartirii primului cu al doilea polinom va fi afisat in casuta.
5. Derivare: Rezultatul derivarii primului polinom si optional celui de-al doilea vor vi afisate separate prin “||”.
6. Integrare: Rezultatul integrarii primului polinom si optional celui de-al doilea vor vi afisate separate prin “||”.

Proiectare



Structura de date folosita este HashMap-ul. Acesta este o structura de date care stocheaza elemente in perechi cheie-valoare si permite accesarea si modificarea acestora in timp constant, bazat pe o functie de hash care transforma cheia intr-un index intr-un tabel de dispersie.

Pentru operatii s-au folosit urmatorii algoritmi:

𝑃(𝑋) + 𝑄(X) = (𝑎𝑛+𝑏𝑛) ∗ 𝑋^𝑛 + (𝑎𝑛−1+𝑏𝑛−1) ∗ 𝑋^(𝑛−1) + ⋯ + (𝑎1+𝑏1) ∗ 𝑋 + (𝑎0+𝑏0)

𝑃(𝑋) - 𝑄(X) = (𝑎𝑛-𝑏𝑛) ∗ 𝑋^𝑛 + (𝑎𝑛−1-𝑏𝑛+1) ∗ 𝑋^(𝑛−1) + ⋯ + (𝑎1-𝑏1) ∗ 𝑋 + (𝑎0-𝑏0)

Pentru inmultire, se inmulțesc fiecare monom dintr-un polinom cu fiecare monom din celalalt polinom, se adună rezultatele și se simplifica, daca este necesar.

In cadrul operatii de impartire avem urmatorul algoritm.

Pasul 1 - Ordonam monoamele celor doua polinoame P si Q in ordine descrescatoare in functie de gradul lor.

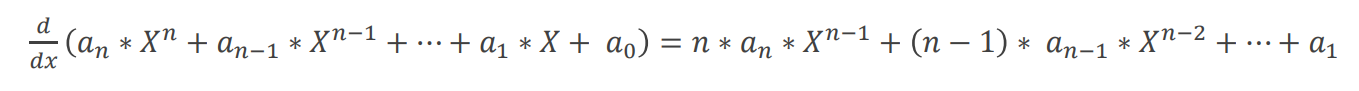
Pasul 2 - Impartim polinomul cu cel mai mare grad la celalalt polinom cu grad mai mic (sa presupunem ca P are cel mai mare grad)

Pasul 3 - Impartim primul monom al lui P la primul monom al lui Q si obtinem primul termen al catului.

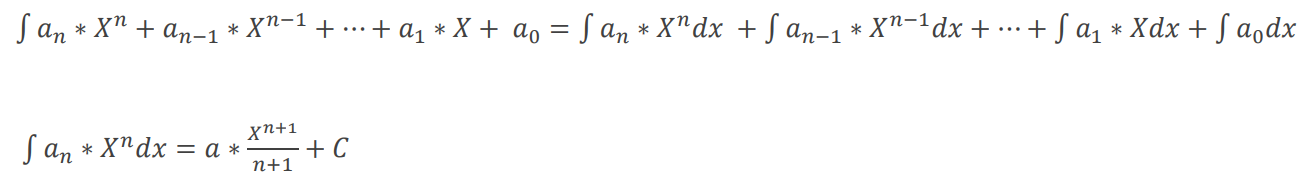
Pasul 4 - Inmultim catul cu Q si scadem rezultatul inmultirii din P, obtinand restul impartirii.

Pasul 5 - Repetam procedura incepand cu pasul 2, luand restul ca dividendul nou al impartirii, pana cand gradul restului este mai mic decat cel al lui Q.

Algoritmul pentru derivare:



Algoritmul pentru integrare:



Implementare

Dupa cum se poate observa in diagrama UML, proiectul este alcatuit din 3 clase:

1. Polinom

Clasa care defineste un polinom, format dintr-un HashMap de monoame. Pe langa lista de monoame, mai avem o variabila de tip bool C, care va fi folosita la operatia de integrare / derivare, pentru a afisa variabila constanta unde este cazul.

Vom avea metodele adaugaMonom, care adauga monomul a\*x^n prin perechea (a, n) in polinom, getMonom, care ne permite sa accesam coeficientul monomului de la un anumit exponent, getGrad, care ne va returna gradul polinomului si toString, care va transforma polinomul intr-un String usor de citit de catre utilizator.

1. Calculator

Aceasta clasa va fi cea in care vom defini operatiile, asadar va consta doar din metodele adunare(), scadere(), inmultire(), impartire(), integrare() si derivare(), in care avem implementati algoritmii descrisi mai sus.

1. Interfata grafica

In aceasta clasa vom avea interfata grafica, construita pe baza Java Swing.

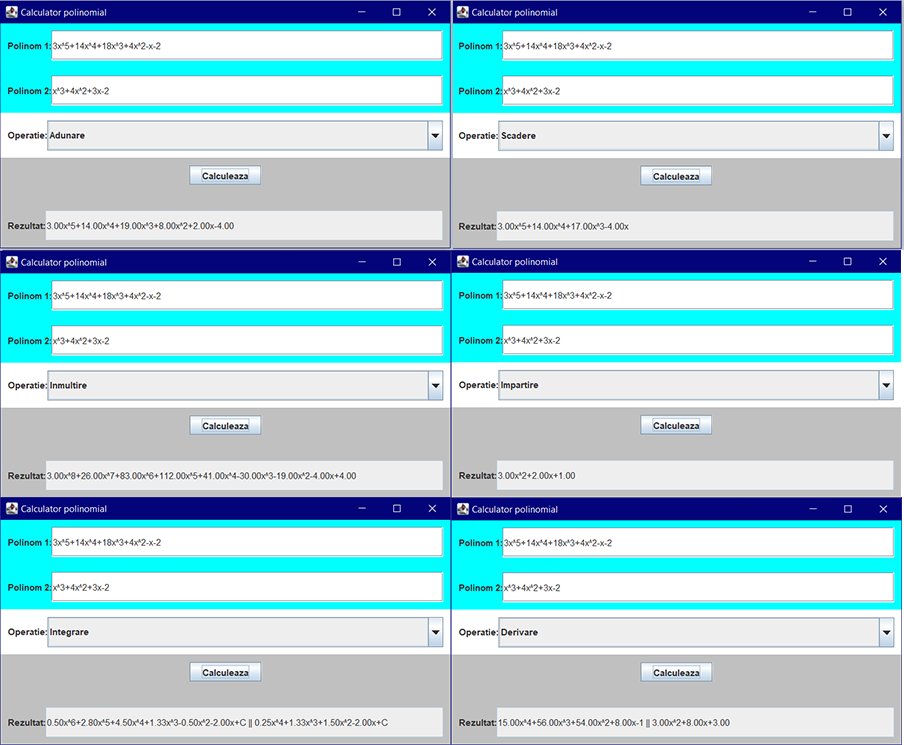
In aceasta putem gasi un JComboBox, folosit pentru a alege operatia, polinom1Field si polinom2Field, casutele pentru introducerea polinoamelor si rezultatField, casuta pentru introducerea rezultatului. In cea ce priveste metodele, acestea sunt urmatoarele:

convert() – aceasta metoda transforma un String (cel introdus de utilizator in casuta) intr-o variabila de tip Polinom folosindu-se de regex. Prima data, cu ajutorul unui regex, polinomul va fi impartit in monoame, iar apoi folosind al regex, monoamele vor fi impartite in coeficient si exponent, iar apoi adaugate in Polinomul proaspat creat.

actionPerformed() – aceasta metoda este inclusa in javaswing si ne permite sa afisam rezultatul operatiei, tot aici apelam metoda convert() pentru a transforma string-urile transmise de utilizator in variabile Polinom, iar apoi in functie de varianta aleasa, afisam in casuta rezultatul corespunzator.

Rezultate

Exemple de operatii:



Concluzii

Pe parcursul rezolvarii acestei teme am invatat / imbunatatit urmatoarele aspecte: crearea de interfete folosing javaswing, utilizarea regex, utilizarea hashmap si multe altele. In ceea ce priveste posibile imbunatatiri, calculatorul poate primi o interfata grafica cu un design mai modern, mai elegant, pot fi implementate si alte operatii precum ridicarea la putere sau gasirea de radacini.

Bibliografie

1. <https://dsrl.eu/courses/pt/> - cerinte, prezentari, exemple

2. <https://www.javatpoint.com/java-swing> - tutorial javaswing