MINISTERUL EDUCAȚIEI ȘI CERCETĂRII ȘTIINȚIFICE

UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

DOCUMENTATIE TEMA 1

CALCULATOR DE POLINOAME

Trifu Diana-Maria
Grupa 30223
Profesor Laborator: Dorin Moldovan

Cuprins:

1. Cerințe funcționale

Propuneți, proiectați și implementați un sistem de procesare a polinoamelor de o singură variabilă cu coeficienți intregi, alături de o interfață grafică prin intermediul căruia utilizatorul va putea performa următoarele operații, afișând și rezultatul acestora:

- **Adunarea a două polinoame**
- **Scăderea a două polinoame**
- **Înmulțirea a două plinoame**
- **Împărțirea a două polinoame**
- **Derivarea unui polinom**
- **Integrarea unui polinom**

Considerații de implementare:

- **Folosirea limbajului de programare Java**
- **Folosirea Java Swing pentru implementarea interfeței grafice**
- **Regex – pentru verificarea validității polinoamelor**
- **Folosirea listelor în locul vectorilor**
- **Folosirea foreach în locul for(int i = 0…)**
- **Implementarea claselor trebuie să fie cu maximum 300 de linii de cod(cu excepția claselor UI)**
- **Implmentarea metodelor trebuie să aibă maximul 30 de linii de cod**
- **Folosirea JUnit pentru testarea aplicației**

2. Obiectivul temei

2.1 Obiectivele secundare

Obiectivul principal al proiectului este de a creea o aplicație care să pună la dispoziție utilizatorului un sistem de calcul al polinoamelor, prin intermediul unei interfețe grafice. Polinoamele sunt formate din unul sau mai mulți termeni numiți monoame, care sunt alcătuite dintr-o constantă, numită coeficient, înmulțită cu o variabilă, fiecare variabilă putând avea un exponent constant, întreg și pozitiv.

2.2 Obiectivele secundare

Obiectivele secundare ale temei reprezintă pașii careau fost urmați pentru atingerea obiectivului final.

1. **Alegerea structurilor de date și a claselor folosite pentru dezvoltarea proiectului, pentru atingingerea obiectivului final al proiectului. Folosirea unui model MVC(Model-View-Controller) pentru a putea forma o GUI(Graphic User Interface), care permiteinteracțiunea utilizatorului cu calculatorul de polinoame.**
2. **Dezvoltarea metodelor/ algoritmilor – se descriu algoritmii care permit realizarea operațiilor cu polinoame, într-o forma cât mai simplă și eficientă, astfel încât metodele nu vor depași 30 de linii de cod.**
3. **Implementarea soluției finale – se vor descrie clasele de care este nevoie pentru reprezentarea polinoamelor, dar și cele care fac posibilă realizarea interfeței grafice, alături de câmpurile și metodele importante.**
4. **Testare – Vor fi descrise câteva scenarii de testare a operațiilor pe polinoame, folosindu-ne de JUnit Test.**

2.3 Scenarii

Descrierea operației de adunare a două polinoame

Pasul 1: Utilizatorul va introduce de la tastatură în câmpul text pentru polinomul 1(imediat sub label-ul care indică "Primul polinom") un polinom care are termenii de forma "coeficientx^grad", se va proceda identic și pentru polinomul al doilea care va fi introdus în câmpul text din dreptul label-ului "Al doilea polinom".

Pasul 2: Utilizatorul va selecta operația matematică pe care dorește să o efectueze cu cele două polinoame, alegând unul dintre câmpurile unui Combo-Box, acestea conținând în mod explicit numele operatiilor matematice pe care aplicatia le poate executa, adică: adunare, scădere, împărțire, înmulțire, derivare sau integrare. În cazul care este descris acesta ar trebui să aleagă câmpul în care este scris "adunare". În cazul operațiilor matematice realizate pe un singur polinom adică derivarea si integrarea, în cazul în care utilizatorul dorește să efectueze o asemenea operație nu este necesară introducerea a două polinoame, deoarece operația se va executa pe polinomul din primul câmp text(imediat sub label-ul „Primul Polinom").

Pasul 3: După apăsarea butonului corespunzator operației, aplicația va lua în considerare doar termenii introduși conform formatului menționat mai sus, ignorând textul introdus de utilizator ce nu corespunde cu formatul specificat. În cazul în care în unul din câmpurile text nu este identificat niciun termen introdus corect sau nu există nimic introdus se va deschide o nouă fereastră care va atenționa utilizatorul ca datele introduse nu respectă formatul și operația selectată nu se poate executa.

Pasul 4: În cazul în care datele sunt introduse corect, programul va genera rezultatul dorit și îl va afișa în ultimul chenar de text(sub label-ul „ Rezultat: ").

Pasul 5: Polinoamele introduse inițial rămân în câmpurile text, astfel fiind posibilă efectuarea unei alte operații pe aceleași două polinoame.

În cazul celorlalte operații pe care aplicația le pune la dispoziție, pașii urmăriți sunt aceeași, ceea ce diferă fiind doar alegera din cadrul Combo-Box-ului.

3. Analiza Problemei

Utilizarea interfeței presupune introducerea de catre utilizator în două JTextField-uri polinoamele care urmează să fie supuse efectuării unor operații(adunare, scădere, înmulțire, împarțire, derivare sau integrare). Alegerea operației ce urmează a fi efectuată asupra celor două polinoame se poate alege prin intermediul unui JComboBox, ar dupa apasarea butonului "=", rezultatul va fi afișat într-un al treilea JTextField.

Deoarece este posibil să se introducă date greșite(String-uri care nu reprezintă un polinom), se va afișa un mesaj pe ecran care va avertiza utilizatorul că datele introduce de el sunt invalide(nu respectă formatul). Dacă polinomul introdus este dezordonat(gradele variabilei nu sunt în ordine descrescătoare, acesta va fi considerat corespunzător, deoarece polinomul va fi aranjat ulterior de către program, înaintea afișării rezultatului așteptat). În cazul derivării sau al integrării, rezultatul afișat va fi cel pentru primul polinom introdus de către utilizator.

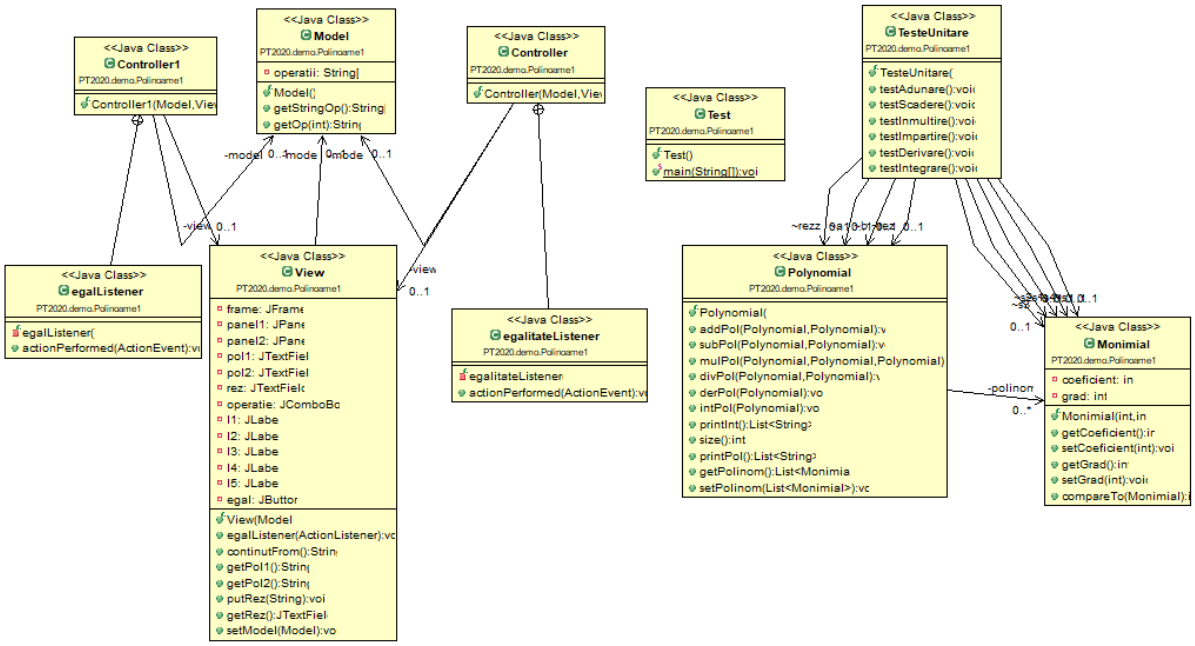
4. Proiectare

4.1. Structuri de date și clasele folosite

Ca și clase principale folosite sunt: Polynomial și Monomial. Clasa Monomial descrie obiectele de tip monom care sunt formate dintr-un coeficient, un grad și o variabilă(2x^3 – unde 2 este coeficientul, x – variabila și 3 puterea/gradul). Această clasa conține ca variabile instanță doi întregi: unul reprezentat de puterea variabilei x, gradul și unul care reprezintă coeficientul. Clasa Polynomial conține un ArrayList, structura de date principală folosită în implementarea acestui proiect, în care sunt stocate mai multe monoame care alcătuiesc un polinom. Celelalte 3 clase principale sunt cele care alcătuiesc interfața grafică a proiectului: Model - View- Controller.

4.2. Diagrama de clase

Unified Modeling Language (prescurtat UML) este un limbaj standard pentru descrierea de modele și specificații software. Diagrama de clase UML este folosită pentru reprezentarea vizuală a claselor și a interdependențelor, taxionomiei și a relațiilor de multiplicitate dintre ele. Diagramele de clasă sunt folosite și pentru reprezentarea concretă a unor instanțe de clasă, așadar obiecte și a legăturilor concrete dintre acestea.



4.3. Algoritmi

Adunare monoame

Acest algoritm presupune generarea unui nou monom care are ca și coeficient suma coeficienților rezultați prin adunarea celor 2 monoame si gradul același ca și gradul celor doua monoame.

Scădere monoame

Acest algoritm presupune generarea unui nou monom care are ca și coeficient diferența coeficienților rezultați prin scăderea celor 2 monoame și gradul același ca și gradul celor două monoame.

Înmulțire monoame

Acest algoritm presupune generarea unui nou monom care are ca și coeficient produsul coeficienților rezultați prin înmulțirea celor 2 monoame și gradul reprezintă suma gradelor celor 2 monoame.

Derivare monom

Derivarea monomului presupune ca primul coeficient să fie produsul dintre coeficientul monomului ce trebuie a fi derivat și gradul acestuia, noul monom va avea gradul egal cu gradul curent minus 1.

Integrare monom

La integrarea monomului coeficientul nou va fi coeficentul vechi împărțit la gradul actual minus 1, iar coeficientul nou este cel actual incrementat cu valoarea 1.

Adunare polinoame

Adunarea a două polinoame este implementată prin parcurgerea primului polinom și parcurgerea celui de-al doilea polinom, pentru fiecare monom din primul. În acest fel, în momentul în care pentru un monom din primul polinom se găsește un monom în cel de-al doilea polinom cu același grad ca și el, coeficienții celor două monoame se vor aduna și vor duce la crearea unui nou monom, care va fi adăugat în polinomul care reprezintă rezultatul. Cele două monoame din cele două polinoame vor fi șterse. La sfârșitul parcurgerii primului polinom(cel luat ca referință), se vor reparcurge polinoamele inițiale pentru a adăuga în polinomul rezultat, monoamele a căror coeficienți nu au corespondent în celălalt polinom. La sfârșit se sortează polinomul rezultat, astfel încât rezultatul să fie în ordinea descrescătoare a gradelor.

Scădere Polinoame

Algoritmul de scădere este implementat similar cu cel de adunare singurele diferențe fiind ca în parcurgerea simultană se folosește operația de scădere implementată în clasa monom, iar la parcurgeera de la final a celui de-al doilea polinom termenii care nu au suferit modificări în urma scăderii sunt adăugați cu coeficentul lor înmulțit cu –1.

Înmulțire Polinoame

Operația de înmulțire a celor două polinoame presupune parcurgerea simultană a celor două polinoame, în cazul inmulțirii nu mai este necesar să se verifice gradul monoamelor deoarece operația de înmulțire se va face între monoame indiferent de gradul lor. Fiecare monom din primul polinom trebuie înmulțit cu fiecare monom din polinomul al doilea. După efectuarea acestei înmulțiri se va utiliza o funcție de regrupare a termenilor cu același grad care e posibil, să rezulte în urma înmulțirii a două polinoame.

Derivare Polinom

Algoritmul de derivare presupune o simplă parcurgere a polinomului ce trebuie derivat în care fiecărui termen i se aplică metoda de derivare concepută în clasa monom.

Integrare Polinom

Identic cu algoritmul de derivare,doar că se aplică operația de integrare a monomului.

5. Implementare

Clasa Monomial – variabile instanță: doi întregi: coeficientul și exponentul variabilei; metodele acestei clase sunt în principal gettere și settere pentru cele două variabile, dar și metoda compareTo, care permite sortarea monoamelor în ordine descrescătoare în ceea ce privește exponentul variabilei x.

Clasa Polynomial – variabile instanță: un ArrayList de monoame, care alcătuiește polinomul; metodele principale ale acestei clase sunt 6 care descriu modul în care se vor efectua cele 6 operații asupra

celor două polinoame; alte doua metode importante sunt cele care ajută la printarea rezultatului final, în urma unei operații efectuate.

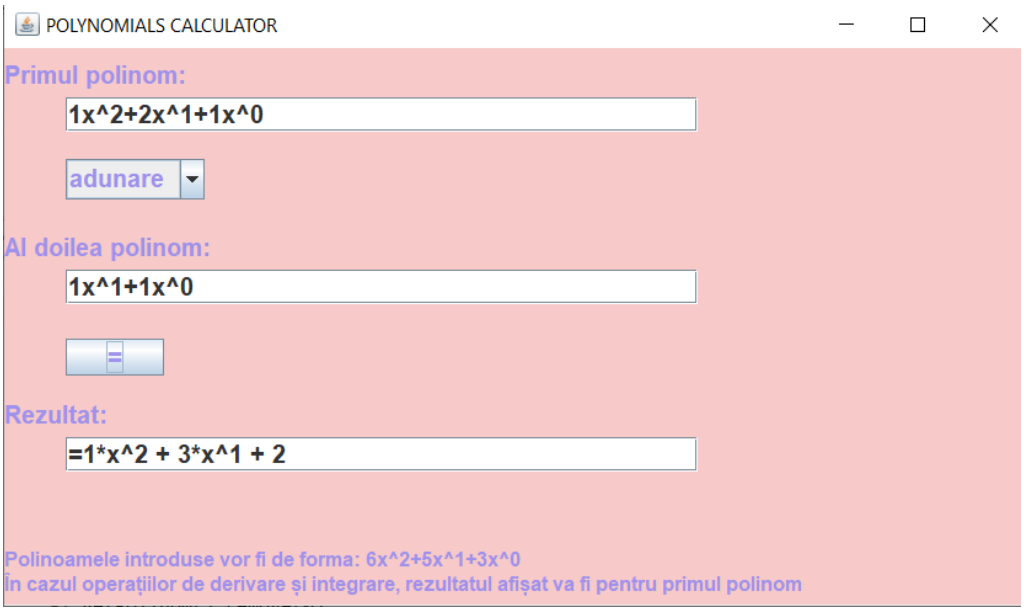
Clasa Model – variabile instanță:un array de string-uri în care se regăsesc numele operațiilor pe care aplicația de calcul a polinoamelor le pune la dispoziție; metodele implementate în cadrul acestei clase sunt reprezentate de două getter-e: unul care preia întreg șirul, iar cealalta preia câte un singur string + clasele Polynomial și Monomial(care reprezintă în mod direct modelul datelor cu care lucrează și pe care le modelează metodele de calcul a polinoamelor din clasa Polynomial).

Clasa View – variabile instanță: frame-ul, două panel-uri, 3 JtextField-uri, un JcomboBox, 5 Jlabel-uri și un Jbutton. Frame-ul este împărțit în două panel-uri în care sunt ogranizate celelalte componente. Cele 3 TextField-uri sunt folosite fie pentru introducerea datelor în interfața grafică de către utilizator, cât și pentru afișarea rezultatului așteptat în urma realzării operației selectate. ComboBox-ul este utilizat pentru selectarea operației dorite, iar Label-urile sunt folosite pentru a indica utilizatorului ce, cum și unde trebuie introduse datele de intrare.

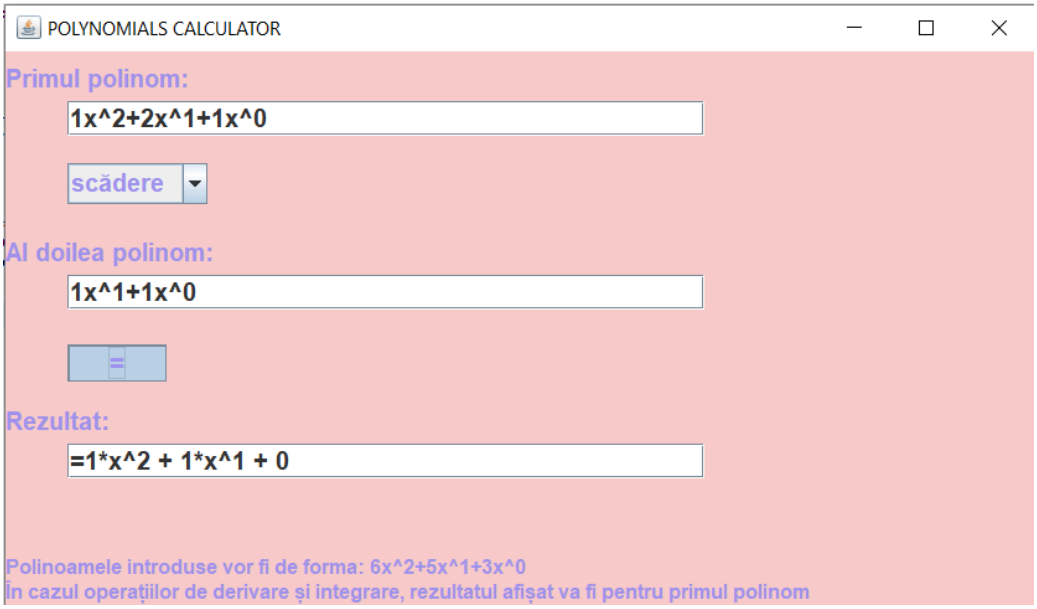
Clasa Controller – variabile instanță: un model și un view; singura metedoă implementată în această clasă este aceea care controlează ceea ce se întamplă cand are loc un eveniment extern asupra butonului "=", în funcție de operația aleasă prin intermediul JcomboBox-ului.

6. Rezultate

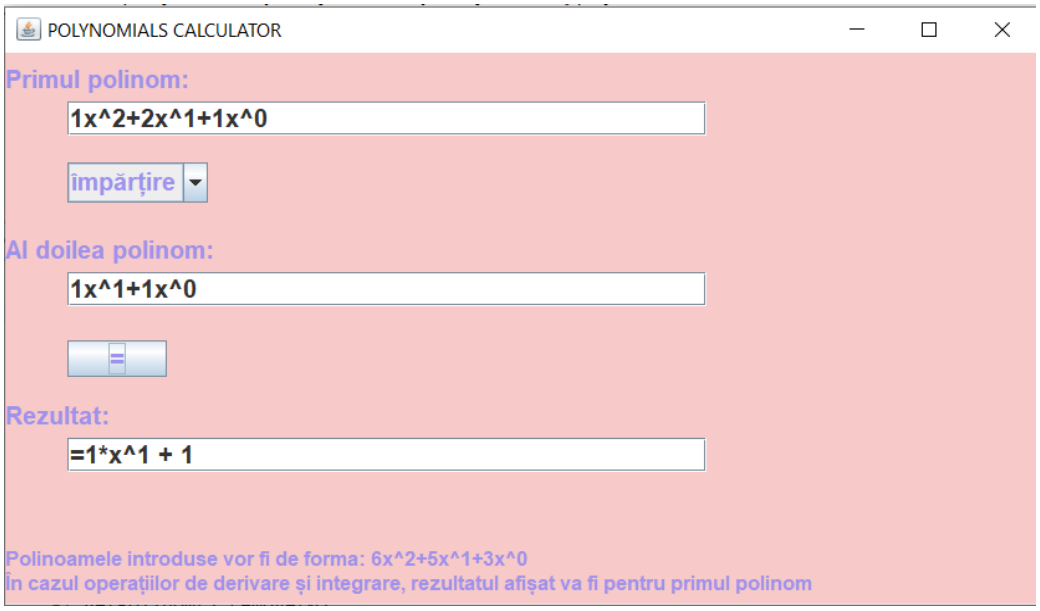
Adunarea a două polinoame



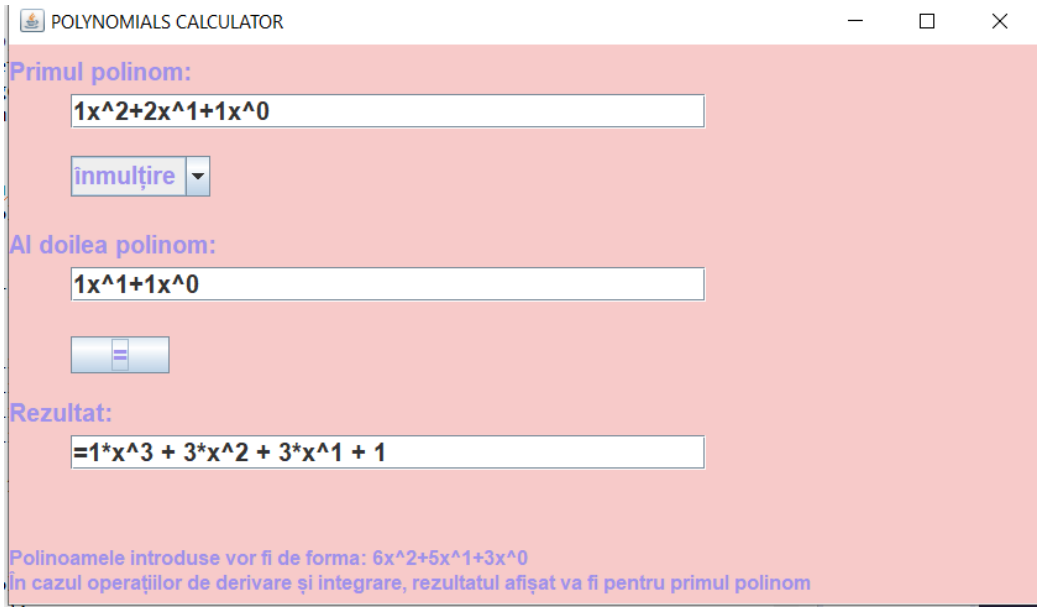
Scăderea a două polinoame



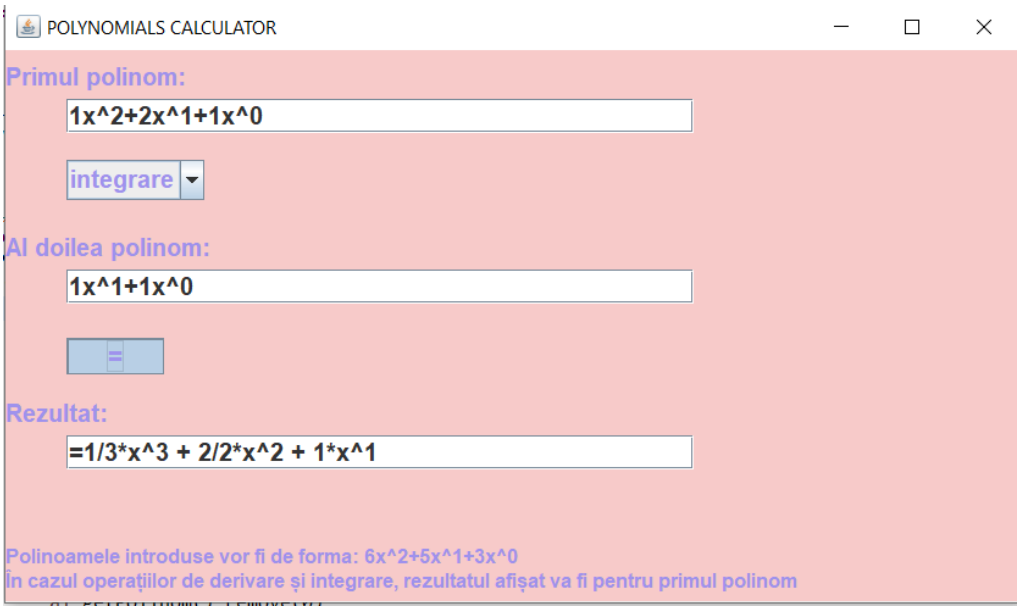
Împărțirea a două polinoame



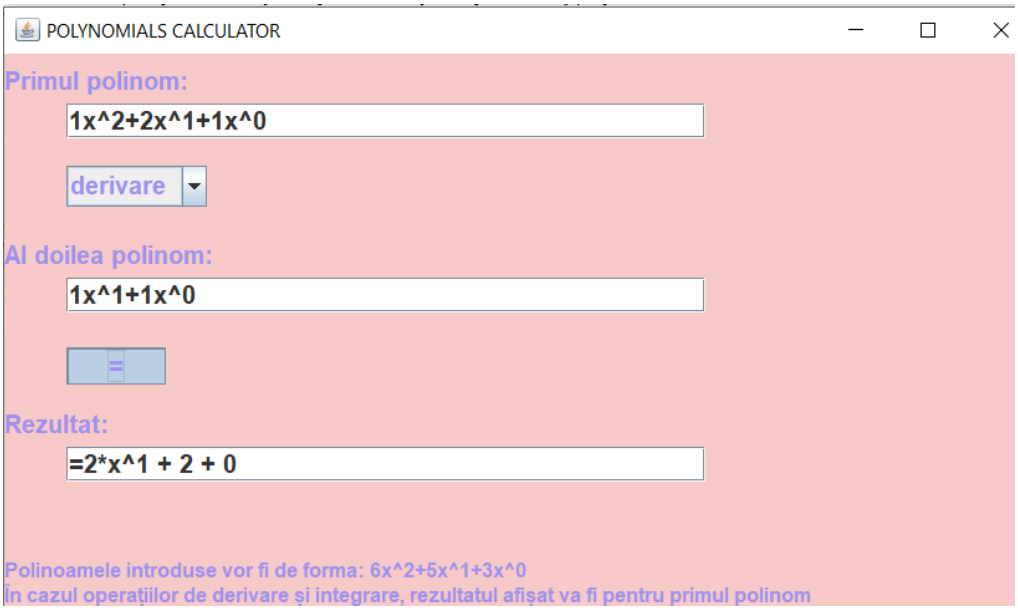
Înmulțirea a două polinoame



Integrarea a două polinoame



Derivarea a două polinoame



7. Concluzii

În concluzie realizarea acestui proiect a fost foarte utilă , amintind de o mare parte din principalele concepte ale porgramării orientate pe obiecte învățate semestrul trecut.

O dezvoltare ulterioară care ar putea fi făcută pentru ca aplicația să fie mai ușor de utilizat ar putea fi spre exemeplu atenționarea mult mai clară a utilizatorului în cazul în care datele sunt introduse greșit, adică programul să îi spună utilizatorului exact care dintre polinoame a fost introdus incorect și să vină cu o sugestie de corectare. În plus, acesta ar fi mai practice și mai util în cazul în care aplicația ar putea permite introducerea corectă a polinoamelorîn mai multe moduri, spre exemplu: "3x^2+2x^1+1x^0" să fie considerat echivalent cu următoarele: "3x^2 + 2x^1 + 1x^0" sau "3x^2+2x^1+1" sau "3 * x^2 + 2 * x^1 + 1 * x^0" sau diferite alte combinații fără a se tine cont de spațiile libere lăsate sau alte semne în plus(cu * sau fără). Alte dezvoltari ale aplicatiei ar putea fi realizarea de operatii succesive sau posibilitatea de a se efectua operații pe mai mult de două polionoame. În ceea ce privește interfața grafică , s-ar putea face un meniu principal care să se deschidă la rularea aplicației și care să aibă mai multe butoane care să deschidă frame-uri separate pentru fiecare operație disponibilă în parte. Astfel fiecare operație ar putea fi individualizată, iar frameurile respective ar putea conține mai multe informații despre modul în care se realizează operația respective pe polinoame, despre modul în care ar trebui introduse datele astfel încât acestea să fie validate de către aplicație, iar rezultatul să fie introdus correct.

8. Bibliografie

-
- **https://www.baeldung.com/java-copy-on-write-arraylist**
- **https://www.vogella.com/tutorials/JavaRegularExpressions/article.html**
- **https://stackoverflow.com/questions/28859919/java-regex-separate-degree-coeff-of-polynomial**
- **http://www.mkyong.com/tutorials/junit-tutorials/**
- **http://ptgmedia.pearsoncmg.com/images/9780321927767/samplepages/03219277 61.pdf**