**DOCUMENTATIE TEMA1**

**OPERATII CU POLINOAME**

Nume si prenume:Pop Cristian

Grupa: 30223

**Cuprins**

1.Cerinta

2.Obiective

3.Analiza problemei

4.Proiectare

5.Implementare

6.Testare

7.Concluzii si dezvoltari ulterioare

**1.Cerinta**

Propuneti, proiectati si implementati un sistem de procesare a polinoamelor de o singura variabila cu coeficienti intregi.

**2.Obiective**

Obiectivul principal este crearea de polinoame si implementarea unor operazii de baza asupra lor cum ar fi adunarea,scaderea,inmultirea,impartirea,derivarea si integrarea acestora.La aceste obiective principale se adauga si obiectivele secundare prezente in tabelul de mai jos.

|  |  |  |
| --- | --- | --- |
| **Obiectiv Secundar** | **Descriere** | **Capitol** |
| Dezvoltarea de use case-uri si scenarii |  | 3 |
| Alegerea structurilor de date |  | 4 |
| Impartirea pe clase |  | 4 |
| Dezvoltarea algoritmilor |  | 4 |
| Implementarea solutiei |  | 5 |
| Testare |  | 6 |

**3.Analiza problemei**

Putem defini un user care va interactiona cu programul pe care l-am creat.Acesta isi creeaza in primul rand un monom(ex 2\*X^3) care va fi adaugat automat intr-un polinom.Monoamele vor fi identificate prin doua campuri:coeficient si grad care vor fi alese de user.Acestea trebuie sa fie de tip integer.Polinomul va fi creat in functie de nr de monoame pe care userul le introduce.Astfel el va crea doua polinoame asupra carora va putea efectua operatii.

Rezultatul oepratiilor va fi afisat intr-o noua casuta in functie de operatia aleasa.Aceste operatii se vor putea alege cu ajutorul unor butoane.Pentru opeartiile de derivare si integrare este sufficient sa folosim doar unul din cele doua polinoame.

Dupa cum am spus mai sus definim un user care este cel care foloseste interfata.Operatiile pe care le poate face userul sunt crearea de polinoame,adunare acestora ,scaderea acestora,inmultirea acestora,derivarea primului polinom si integrarea primului polinom.

Userul va sti cum vor arata polinoamele.

Sistemul va sti daca datele introduse de user sunt gresite(introducere de litere)

Sistemul va afisa polinomul construit de user sortat

Userul va confirma ca polinoamele create sunt sortate.

Userul va sti care operatie a fost selectata.

Userul va sti asupra carui polinom se vor efectua operatiile de derivare si integrare.

Sistemul va prezenta rezultatele pentru fiecare operatie pe care userul o selecteaza.

Userul va putea sterge rezultatul afisat de o operatie pentru a testa alta operatie si a nu suprascrie peste rezultatul anterior.

Userul va inchide sistemul.

**4.Proiectare**

Pentru a putea proiecta aceasta aplicatie cu success avem nevoie de 3 clase:Monom,Monom1,Main.

Cea mai simpla si practica metoda mi sa parut sa construiesc polinoamele cu ajutorul monoamelor.Astfel in clasa Monom stocam coeficientul si gradul monomului,care sunt de tip integer, dupa ce acestea au fost introduse in casete si adaugate in polinom.Tot in aceasta clasa am implementat operatiile de adunare,scadere,inmultire si derivare.Aceste operatii lucreazacu obiecte de tip monom(Mon1.adunare(Mon2),Mon1.scadere(Mon2),Mon1.inmultire(Mon2).Mon1.derivare).Pentru a adauga elementele introduce in casete in monoame avem nevoie sa convertim aceste valori introduse in caseta(care sunt de tip String) in integer,iar apoi ele vor fi adaugate in monoame.

In clasa Monom1 stocam coeficientul si gradul monoamelor,care de aceasta data sunt de top double pentru a ne ajuta la integrare.

In clasa Main ne cream interfata si transmitem fiecare cerinta a utiliatorului catre softul creat.Interfata create este usor de utilizat pentru user,ea continand 7 JTextFielduri in care se introduc coeficientii si gradele polinoamelor(4),se afiseaza polinoamele(2) si se afiseaza rezultatul operatiilor(1).De asemenea mai exista 8JButtonuri,pentru adaugarea in polinoame a monoamelor(2),pentru operatii(5) si pentru curatarea textfieldului “Rezultat”.Tot in Main ne cream ArrayListuri pt polinoame in care stocam monoamele introduse in interfata(astfel ne cream polinoamele si le avem salvate).

Algoritmi folositi la implementarea operatiilor:

In cazul adunarii si scaderii se va rezuma la adunarea coeficientilor monoamelor,cat timp gradul ramane egal.Daca nu se adauga monomul cu gradul mai mare pana cand toate elementele polinoamelor sunt introduse in suma.Acelasi algoritm se foloseste si la scadere.Pentru a sti care monom se adauga se foloseste o functie de comparare care compara gradele si il allege pe cel cu grad mai mare.Apoi se deplaseaza la elemental urmator din acel polinom.

In cazul inmultirii se va rezuma la inmultirea coeficientilor monoamelor si adunarea gradelor acestora.Aceasta operatie se face pentru fiecare pereche de 2 monoame cu un monom din primul polinom,iar al doilea monom din al doilea polinom.

In cazul derivarii se inmulteste coeficientul monomului cu gradul acestuia,iar gradul se scade.

In cazul integrarii se imparte coeficientul monomului cu inversul dintre grad plus unu,iar gradul se aduna cu unu.Pentru integrare este necesar ca si coeficientul si gradul monomului sa fie de tip double pentru a se putea face impartirea cum trebuie.

Pentru ca rezultatul acestor operatii sa fie afisat pe textfield-ul “Rezultat” este necesar ca ArrayListurile care contin rezultatele operatiilor sa fie adaugate intr-o variabila de tip String cu ajutorul functiei concat.

Interfata contine urmatoarele clase predefinite:

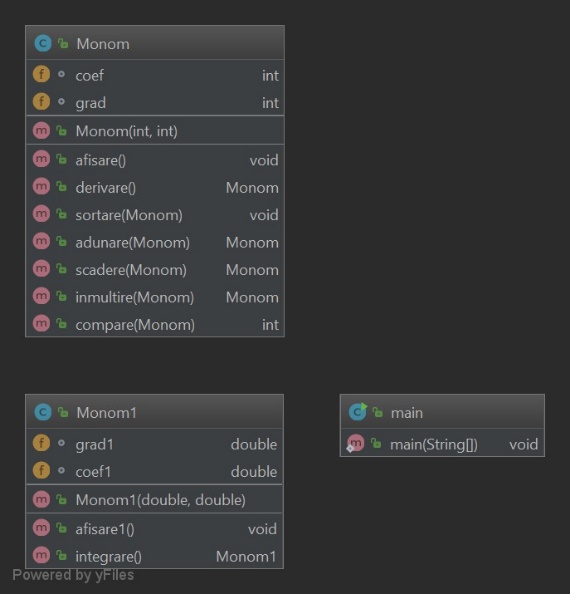
JButton:cand se apasa un anumit buton se executa o anumita instructiune

JLabel:afiseaza text in interfata fara a putea fi modificat din aceasta

JTextField:o caseta in care se pot introduce si afisa date

JPanel:fereastra principala in care se adauga componentele enumerate mai sus

**Diagrama UML**

****

**5.Implementare**

Proiectul contine 3 clase.Prima clasa este cea de “Monom” care este alcatuita din coefficient si grad,ambele de timp integer.Metodele pe care aceasta clasa le contine sunt operatiile asupra monoamelor.Prima metoda este cea de adunare in care este definit un monom nou in care se va scrie rezultatul adunarii a doua monoame.Se initializaeaza monoamele dupa care se verifica daca gradele lor sunt egale,caz in care coeficientii se aduna si se returneaza in monomul nou impreuna cu gradul oricaruia din cele doua monoame.Daca gradul primului este mai mare decat gradul celui de al doilea monom se returneaza primul monom,iar daca este mai mic se returneaza al doilea monom.

A doua metoda este operatia de scadere,in care se repeat pasii de la adunare,singura schimbare fiind ca atunci cand gradele sunt egale coeficientii se scad.

A treia operatie este cea de inmultire.Aici nu trebuie sa verificam gradele monoamelor,ci doar trebuie sa returnam un monom nou in care inmultim coeficientii monoamelor si adunam gradele.Daca gradul monomului returnat este 0 atunci monomul rezultat va contine doar coeficientii inmultiti,iar daca produsul coeficientilor este 0 se va returna 0.

A patra operatie este cea de derivare.Aici avem nevoie de un singur monom.Returnam un alt monom in care avem produsul dintre coeficientul si gradul primului monom si gradul acestuia minus unu.Derivarea se face atata timp cat gradul este diferit de 0.Daca este egal cu 0 se returneaza 0.

Pe langa aceste operatii mai este o metoda de comparatie care returneaza 2 daca gradul primului monom este mai mare decat gradul celui de al doilea monom,1 daca gradul celui de al doilea monom este mai mare decat gradul primului monom si 0 daca gradele celor doua monoame sunt egale.Avem nevoie de aceasta clasa in main pentru operatiile de adunare si scadere.

Ultima metoda prezenta in aceasta clasa este cea de sortare care sorteaza doua monoame cu ajutorul interschimbarii atat a gradului cat si a coeficientului pt ca polinomul creat sa fie sortat indifferent de ordinea in care sunt introduse monoamele.

A doua clasa este denumita Monom1 si este folosita pentru a declara un monom cu coefficient si grad de tip double.Avem nevoie de aceasta clasa pentru a implementa operatia de derivare,care returneaza un monom care contine produsul dintre coeficient si inversul gradul plus unu si gradul monomului minus unu pe care vrem sa il derivam.

Ultimul element al acestui proiect este programul principal,adica mainul in care ne construim interfata grafica si aplicam operatiile enumerate mai sus.Interfata grafica este alcatuita dintr-un JFrame care contine un JPanel une sunt implementate toate componentele necesare pentru aceasta interfata.In acest JPanel exista doua JLabel denumite coef,doua JLabel denumite grad,4JTextField in care se introduc coeficientii si gradele monoamelor.Astfel primele doua sunt pentru primul polinom si celelalte pentru urmatorul polinom.De asemenea sunt doua butoane JButton denumite Add care au nroulul de a introduce monomul in polinom,care este alcatuit dintr-un JLabel Polinom si un JTextField unde se va afisa polinomul creat.De fiecare data cand vom apasa add se va introduce elemental creat in functie de ce introducem in casete in polinom.Prin urmare vom avea in cele doua casete de mai jos de coeficienti si grade polinoamele create si asupra carora vom efectua operatiile.Dedesubtul acestor polinoame avem butoanele care corespund fiecarei operatii:adunare,scadere,inmultire,derivare,integrare.Apasand oricare din acele butoane mai jos intr-un alt JTextField denumit “Rezultat” se va afisa rezultatul acestor operatii.Se vor aduna ,scadea(al doilea polinom din primul) si inmulti cele doua polinoame;derivarea si integrarea aplicandu-se doar pentru primul polinom.Ultimul element existent in aceasta interfata este butonul “Clear” care sterge cee ace se afla in rezultat.

Tot in main ne declaram cateva ArrayList-uri de monoame,acestea desemnand polinoame.Fiecare monom introdus va fi introdus intr-un ArrayList,astfel formandu-se polinoamele.Avem ArrayListurile p si q care vor fi polinoamele formate de noi.Mai avem nevoie de ArrayListurile suma,dif,produs in care vom introduce rezultatele operatiilor de adunare,scadere si inmultire.De asemenea in ArrayListurile derivare si integrare vom avea primul polinom derivate si integrat.

Pentru fiecare din butoanele enumerate mai sus avem nevoie de o functie ActionListener pentru ca butoanele sa faca operatiile pentru care sunt definite.Pentru butoanele de add vom avea nevoie de doua stringuri pentru a stoca coeficientii si gradele introduse in caseta(cee ace introducem in caseta este de tip String),iar apoi in alte variabile de tip integer vom converti aceste Stringuri in integer cu ajutorul functiei Integer.parseInt.Ne cream un monom nou in care introducem aceste valori,iar acest monom va fi adaugat in ArrayListurile p sau q(depinde pt care polinom introducem) cu ajutorul functie add(p.add(m)).Astfel in functie de cate date introducem cu ajutorul butoanelor add ne cream polinoamele.Tot aici intervine functia de sortare din clasa Monom care inainte sa aduge in ArrayListuri verifica gradul monomului introdus cu gradele celorlalte monoame si astfel polinomul obtinut este deja sortat fiind mai usor de implementat operatiile,in special cele de adunare si scadere.

Pentru operatiile de derivare si integrare ne vom folosi de ArrayListul p care contine primul polinom.Functiile pt aceste butoane parcurg polinomul,iar pentru fieacre element al sau care este un monom apelam functiile de derivare(din clasa Monom) si integrare(din clasa Monom1).Vom avea nevoie de o variabila de tip String in care vom introduce fiecare element al polinomului dupa efectuarea operatiei.Astfel la final polinomul rezultat va fi de tip String,iar cu ajutorul functiei rezultat11.setText(p)(rezultat11 este JTextFieldul in care afisam rezultatul).

Pentru operatiile de adunare si scadere vom avea nevoie pe langa elementele enumerate la derivare si integrare si de doua variabile de tip integer care vor fi initializate cu 0 si vor reprezenta pozitia monoamelor din cele doua polinoame.Cele doua polinoame vor fi parcurse cu un while si in cuntie de rezultatul functie de sortare(daca este 2 adaugam in suma primul el din primul monom,daca este 1 adaugam primul element din al doilea monom,iar daca este 0 adaugam suma monoamelor de pe pozitiile curente ale polinoamelor).In functie de ce adaugam crestem constantele count1 si count2;count1 pt primul polinom,count2 pt al doilea polinom si ambele pt suma.Mai avem nevoie de 2 while-uri in caz ca lungimile polinoamelor sunt diferite;astfel cu aceste while-uri vom adauga in suma elementele ramase dinntr-un polinom.Aceasta metoda este valabila si pentru scadere.

Pentru operatia de inmultire parcurgem polinoamele p si q cu doua for-uri si apelam functia de inmultire pe elementele curente.Afisarea se face exact dupa modelul de la adunare.

Ultima functie pentru butoul Clear sterge cee ace se afla in rezultat11.

**7.Concluzii si dezvoltari ulterioare**

In concluzie,aceasta aplicatie poate fi utilizata de client pentru a efectua diferite operatii cu polinoame.

Cee ace am invatat din aceasta tema este faptul ca trebuie sa analizez prima data problema,sa o impart in subprobleme si apoi sa o implementez cu ajutorul codului.Este mai bine sa ne gandim la o solutie mai intai sis a vedem toate cazurile posibile,iar apoi sa o implementam decat sa implementam de la inceput si apoi sa corectam greselile aparute sau cazurile nerezolvate.

Cee ace poate fi imbunatatit la aceasta aplicatie este eficienta anumitor algoritmi in special a celui de sortare unde s-ar fi putut folosi functia swap.Se mai poate imbunatati eficienta algoritmului de inmultire.Se mai poate introduce operatia de impartire.

O alta dezvoltare care ar ajuta clientii ar fi ca fiecare buton cu operatia cand este apasat sa deschida o pagina web cu exemple despre cum se efectueaza operatia.Astfel utilizatorii pot invata aceste operatii.O alta operatie care ar putea fi implementata ar fi ridicarea la putere a unui polinom sau chiar si aflarea radacinilor polinomului.