DOCUMENTATIE

TEMA *NUMARUL\_TEMEI*

NUME STUDENT: …………………

GRUPA: …………………

# CUPRINS

[1. Obiectivul temei 3](#_Toc95297885)

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

[3. Proiectare 3](#_Toc95297887)

[4. Implementare 3](#_Toc95297888)

[5. Rezultate 3](#_Toc95297889)

[6. Concluzii 3](#_Toc95297890)

[7. Bibliografie 3](#_Toc95297891)

# Obiectivul temei

Este dorită elaborarea unei aplicații dedicate manipulării polinoamelor, o componentă esențială în studiul matematicii și în numeroase domenii tehnice și științifice. Proiectul impune respectarea unor etape esențiale pentru a asigura dezvoltarea eficientă și eficace a aplicației.

1. Proiectarea Algoritmilor de Prelucrare a Polinoamelor:

Procesul de proiectare a algoritmilor pentru manipularea polinoamelor constituie prima etapă în dezvoltarea aplicației propuse. Această etapă implică identificarea și definirea operațiilor de bază, adunarea, scăderea, înmulțirea, împărțirea, derivarea și integrarea polinoamelor. În plus, este necesară analiza eficienței și optimizarea algoritmilor pentru a asigura o performanță optimă a aplicației în condiții diverse de utilizare. Informații suplimentare și detaliate sunt furnizate în cadrul Capitolului 2, unde subiectul este explorat în profunzime.

1. Modelarea Tipurilor de Date Utilizate:

Alegerea unor structuri de date adecvate este crucială pentru eficiența algoritmilor implementați. De asemenea, este important să se țină cont de aspectele legate de gestionarea memoriei și de eficiența în timpul de execuție. Informații suplimentare și detaliate sunt furnizate în cadrul Capitolului 2, unde subiectul este explorat în profunzime.

1. Determinarea unor cazuri de utilizare

Use-case-urile sunt scenarii specifice în care utilizatorii vor interacționa cu sistemul nostru și vor utiliza funcționalitățile oferite de acesta. Prin definirea și înțelegerea acestor use-case-uri, putem ghida procesul de dezvoltare și ne asigurăm că aplicația noastră va satisface nevoile utilizatorilor într-un mod complet și eficient. Informații suplimentare și detaliate sunt furnizate în cadrul Capitolului 2, unde subiectul este explorat în profunzime.

1. Crearea unei Arhitecturi Flexibile și Ușor de Dezvoltat:

O arhitectură bine definită și flexibilă reprezintă o condiție prealabilă pentru o dezvoltare rapidă și eficientă a aplicației. Prin adoptarea unor principii de proiectare software, cum ar fi modularitatea, coeziunea și cuplajul redus, se asigură o structură a codului care poate fi ușor extinsă și modificată în viitor, în funcție de cerințele și feedback-ul utilizatorilor. Informații suplimentare și detaliate sunt furnizate în cadrul Capitolului 3, unde subiectul este explorat în profunzime.

1. Implementarea unei Interfețe Grafice Prietenoase cu Utilizatorul:

Interfața grafică reprezintă componenta prin care utilizatorii interacționează cu aplicația și, în consecință, joacă un rol crucial în experiența lor de utilizare. Crearea unei interfețe grafice intuitive și prietenoase, care să faciliteze utilizatorilor accesul și utilizarea funcționalităților oferite de aplicație, necesită o atenție deosebită acordată designului ergonomic, navigației intuitive și feedback-ului vizual. Informații suplimentare și detaliate sunt furnizate în cadrul Capitolului 3, unde subiectul este explorat în profunzime.

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

*Se va prezenta cadrul de cerinte functionale/ non functionale si cazurile de utilizare ca si diagrame si descrieri de use-case. Descrierile use-case-urilor se vor face sub forma unui flow-chart ori sub forma unei liste continand pasii executiei specifici fiecarui use-case.*

## Cerințe Funcționale:

* *Operarea cu Polinoame:*

*Utilizatorii trebuie să poată efectua operații matematice de bază pe polinoame, cum ar fi adunarea, scăderea, înmulțirea și împărțirea.*

*Aplicația trebuie să ofere funcționalitate pentru derivate și integrale ale polinoamelor.*

*Interfața Utilizatorului:*

* *Aplicația trebuie să aibă o interfață grafică intuitivă și prietenoasă pentru a permite utilizatorilor să introducă și să manipuleze polinoamele cu ușurință.*

*Elementele interfeței utilizatorului, cum ar fi butoanele și meniurile, trebuie să fie clar etichetate și ușor de utilizat.*

## Cerințe Non-Funcționale:

* *Performanță:*

*Aplicația trebuie să fie receptivă și să ofere rezultatele operațiilor într-un timp rezonabil, chiar și pentru polinoame mari sau operații complexe.*

* *Fiabilitate:*

*Aplicația trebuie să fie robustă și să gestioneze corect toate cazurile de utilizare, inclusiv situațiile de eroare și excepții.*

* *Portabilitate:*

*Aplicația ar trebui să fie portabilă și să funcționeze pe diferite platforme și sisteme de operare, cum ar fi Windows, macOS și Linux.*

* *Extensibilitate:*

*Arhitectura aplicației ar trebui să fie concepută în mod modular pentru a permite ușor adăugarea de noi funcționalități sau extinderea celor existente în viitor.*

## Cazuri de utilizare

A diagram of a person with circles and text

Description automatically generated

## Proiectarea Algoritmilor de Prelucrare a Polinoamelor:

## Modelarea structurilor de date

# Proiectare

A diagram of a network

Description automatically generatedA diagram of a flowchart

Description automatically generated

*Se va prezenta proiectarea OOP a aplicatiei, diagramele UML de clase si de pachete, structurile de date folosite, interfetele definite si algoritmii folositi (*daca e cazul)

# Implementare

*Se va descrie fiecare clasa cu campuri si metodele importante. Se va descrie implementarea interfetei utilizator.*

# Rezultate

*Se vor prezenta scenariile pentru testare. In cazul in care in cerinta temei se specifica sa se faca testare unitara cu utilitarul JUnit, se vor integra in acesta sectiune rezultate testari cu JUnit.*

# Concluzii

*Se vor prezenta concluziile, ce s-a invatat din tema, posibile de dezvoltari ulterioare.*

# Bibliografie

*Se vor adauga referintele care au fost consultate de student pe parcursul implementarii temei .*

*Exemplu:*

1. *Bruce Eckel, Thinking in Java (4th Edition), Publisher: Prentice Hall PTRUpper Saddle River, NJUnited States, ISBN:978-0-13-187248-6 Published:01 December 2005.*
2. *What are Java classes? -* [*www.tutorialspoint.com*](http://www.tutorialspoint.com)