  Clasa Polinom

  Realizată de:

- Pantazi Octavian

- Stoica Liviu

Dată: 30 mai 2022

Cuprins:

[Introducere 3](#_Toc104828841)

[Descrierea clasei 4](#_Toc104828842)

[Private 4](#_Toc104828843)

[Funcții ajutătoare 4](#_Toc104828844)

[Public 4](#_Toc104828845)

[Descrierea aplicației care folosește clasa 6](#_Toc104828846)

[Funcția care desenează graficul funcției (funcția complexă) 7](#_Toc104828847)

[Scurt manual de utilizare 11](#_Toc104828848)

[Concluzii și posibilele extinderi 13](#_Toc104828849)

[Tips and tricks 14](#_Toc104828850)

[Bibliografie 15](#_Toc104828851)

# Introducere

Clasa polinom a fost concepută pentru a afla cât mai multe despre diverse polinoame. Astfel, ea permite găsirea unor soluții, aflarea monotoniei pe intervale, derivarea, afișarea graficului și multe altele.

Noi am ales această clasă datorită utilizării intense a polinoamelor în matematică si analiză, dar si pentru că acest subiect este foarte important pentru noi.

# Descrierea clasei

## Private

Partea „private” a clasei conține mai multe lucruri:

* Șirul de caractere „exp” – memorează polinomul dat de la tastatura prin citire cu șir;
* Tabloul cu numere reale „coeficient” – memorează coeficienții fiecărei puteri a polinomului;
* Tabloul cu numere reale „coeficient\_derivare” – memorează coeficienții fiecărei puteri a derivatei polinomului;

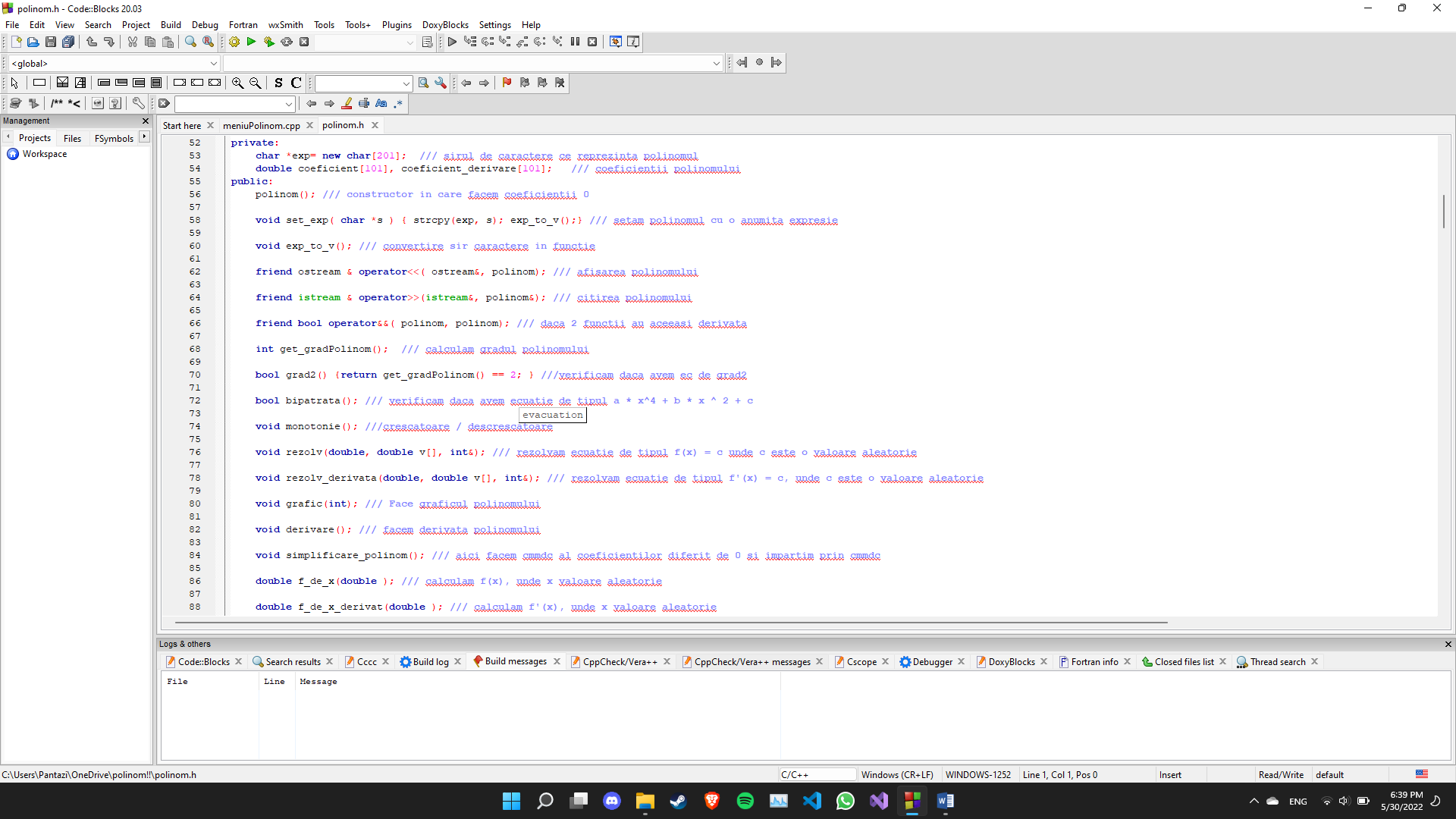
## Funcții ajutătoare

Numeroase funcții ajutătoare precum: putere10( returnează primul parametru înmulțit cu 10 de y ori), Putere(ridicarea la putere), cmmdc(este evident ce face), char\_to\_int (convertește un sir de caractere în număr);

## Public

Partea „public” a clasei conține o multitudine de funcții:

* Constructori( inițializăm polinomul nul – toți coeficienții 0);
* Metode de acces;
* Funcția „exp\_to\_v” – convertește șirul de caractere prin care se poate da polinomul de la tastatura în tabloul cu numere „coeficient”;
* Supraîncărcarea operatorilor pentru citire și afișare, dar și altele precum: operatorul && permite verificarea dacă derivitatele a 2 polinoame sunt egale;
* Funcția „get\_gradPolinom” returnează gradul polinomului;
* Functia „grad2” verifică dacă polinomul este ecuație de gradul 2;
* Funcția „bipatrata” verifică dacă polinomul reprezintă o ecuație bipătrată;
* Funcția „monotonie” oferă numeroase informații despre monotonia polinomului;
* Funcția „rezolv” află soluțiile polinomului;
* Funcția „rezolv\_derivata” află soluțiile derivatei;
* Funcția „grafic” afișeaza graficul polinomului – se va explica pe larg mai târziu;
* Funcția „simplificare\_polinom” caută cmmdc al coeficientilor și simplifică polinomul, dacă este cazul;
* Funcția „f\_de\_x” calculează valoarea functiei într-un punct dat;
* Funcția „f\_de\_x\_derivat” calculează valoarea derivatei într-un punct dat.



# Descrierea aplicației care folosește clasa

Aplicația este foarte utilă și oferă numeroase funcționalități

care ne lumineăza cu o multitudine de informații cu privire la orice polinom introdus de la tastatură. În principiu, aplicația oferă toate functionalitățile explicate în descrierea clasei anterior.

Aplicația constă într-un meniu din care vom avea mai multe opțiuni din care vom alege. Mai jos o să fie atașat o imagine din care se poate vedea cum arată aceasta.

# Funcția care desenează graficul funcției (funcția complexă)

initwindow(getmaxwidth(), getmaxheight(), "Fara nume, asa e mai fancy");

Funcția initwindow, inițializeaza fereastra de desenat cu getmaxwidth() - lungimea maximă monitorului – si cu getmaxheight() - lătimea maximă a monitorului.

line(0, getmaxheight() / 2, getmaxwidth(), getmaxheight() / 2); /// axa Ox

line(getmaxwidth() / 2, 0, getmaxwidth() / 2, getmaxheight()); /// axa Oy

outtextxy(getmaxwidth() - 25, getmaxheight() / 2 + 15, "Ox"); /// afisam Oy pe ecran

outtextxy(getmaxwidth() / 2 - 25, 15, "Oy"); /// afisam Ox pe ecran

Funcția line(int x, int y, int x1, int y1), desenează o linie din punctul (x, y) pana in punctul (x1, y1). În cazul nostru, desenează o linie orizontală (de la mijlocul lățimii din partea stângă până la mijlocul lățimii din partea dreaptă) si alta verticală (de la mijlocul lungimii din partea inferioară până la mijlocul lungimii din partea superioară).

Funcția outtextxy(int x, int y, char\* sir) afișează pe ecranul de desenat textul din variabila “sir”. Cu aceasta afișăm Ox (în partea din dreapta, aproape de jumătatea lățimii) și Oy (în partea de sus, aproape de jumătatea lungimii).

/// sagetile pentru axa Oy

line(getmaxwidth() / 2 - 10, 10, getmaxwidth() / 2, 0);

line(getmaxwidth() / 2 + 10, 10, getmaxwidth() / 2, 0);

/// sagetile pentru axa Ox

line(getmaxwidth(), getmaxheight() / 2, getmaxwidth() - 10, getmaxheight() / 2 - 10);

line(getmaxwidth(), getmaxheight() / 2, getmaxwidth() - 10, getmaxheight() / 2 + 10);

double cnt = 0.1; /// variabila cu ajutorul careia putem sa trasam graficul

setcolor(RED); /// graficul va avea culoarea rosie

setlinestyle(1, 0, 4);

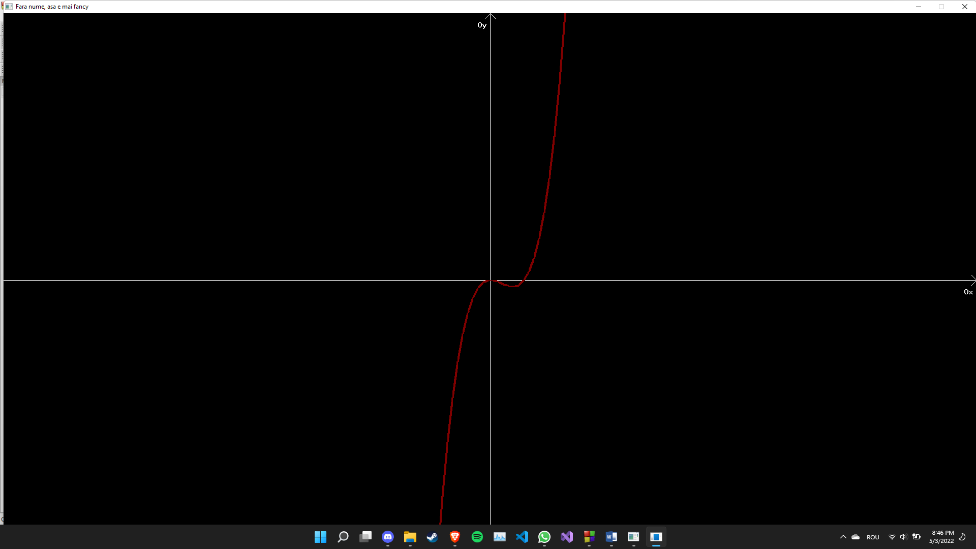
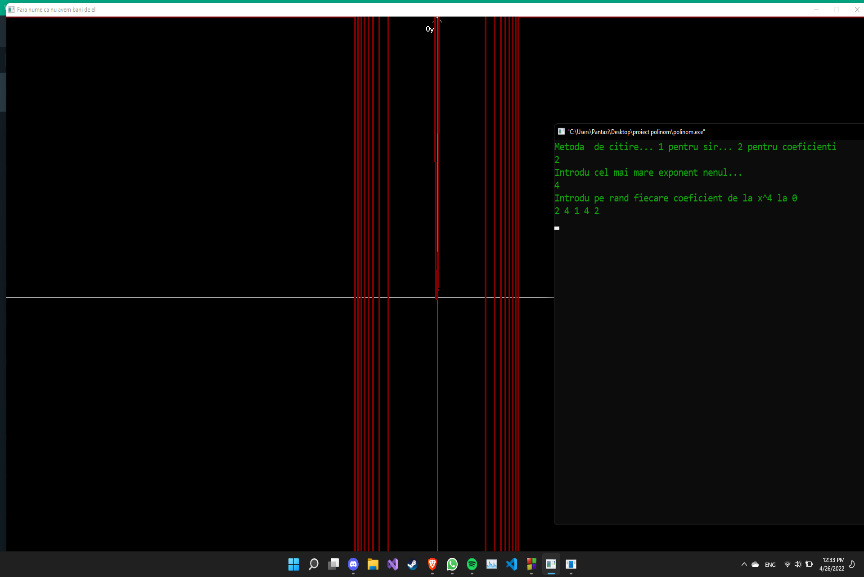
double redimensionare\_segment = 100; /// redimensionez axa Ox

Am facut săgețile la axa Ox și Oy cu două sageți, mai mici, de fiecare.

Variabila “cnt” este folosită pentru a trasa linii din puncte aflate la distanța cnt.

Funcția setcolor(int COLOR) are rolul de a seta o culoare cu care desenăm. Culoarea roșie a fost aleasă, deoarece am considerat ca e potrivita pentru a desena graficul.

Cu funcția setlinestyle(int linestyle, unsigned upattern, int thickness) putem modifica design-ul liniei folosite in desen (<https://home.cs.colorado.edu/~main/bgi/doc/setlinestyle.html>, mai multe informatii despre această funcție se gasește pe acest link).

Variabila “redimensionare\_segment” se folosește pentru a putea face graficul mai mare, mai clar și mai aproape de realitate (Practic am redimensionat segmentele).

În partea stângă avem graficul fără variabila menționată anterior și în partea dreaptă avem cu variabila respectivă.

De la tastatura se va da valorea 1 (dacă utilizatorul vrea să vadă graficul polinomului) sau 2 (dacă utilizatorul vrea sa vadă graficul derivatei polinomului).

for (double i = -(getmaxwidth() / redimensionare\_segment) \* 1.0 + cnt; i <= getmaxwidth() / redimensionare\_segment; i += cnt) {

/// desenam graficul din puncte apropiate (cnt)

line(i \* redimensionare\_segment + getmaxwidth() / 2,

-f\_de\_x(i) \* redimensionare\_segment + getmaxheight() / 2,

(i - cnt) \* redimensionare\_segment + getmaxwidth() / 2,

-f\_de\_x(i - cnt) \* redimensionare\_segment + getmaxheight() / 2);

}

Funcția repetitivă de mai sus, începe de la -(getmaxwidth() / redimensionare\_segment) \* 1.0 + cnt și se termină la getmaxwidth() / redimensionare\_segment, deoarece în acest fel putem să arătăm segmentele obținute mai clar. Ca să putem face redimensionarea este absolut necesar să înmulțim i cu redimensionare\_segment pentru a face segmentul mai mare și astfel devine mai clar. Cum știm de la matematica... (asta in caz ca nu ați uitat) Gf este alcătuită din valori ale funcției. (adică pe grafic va apărea f(1), f(2), f(3), ...). f\_de\_x(double valoare) este o funcție prezentată anterior (analog pentru graficul derivatei).

setcolor(WHITE);

setlinestyle(1, 0, 1);

Am revenit la culoarea albă, pentru a putea desena din nou axele Ox și Oy. Dacă nu făceam acest lucru, programul desena cu culoarea roșie (proprietăți ale funcțiilor - setcolor(int COLOR), setlinestyle(int linestyle, unsigned upattern, int thickness) – odată ce le-am apelat, modificările pe care le-am adus vor rămâne așa ori până când vom apela încă o dată funcțiile sau dacă programul se va termina.

int x, y; ///Se asteapta click stanga pentru a inchide graficul

do{

getmouseclick(WM\_LBUTTONDOWN, x, y);

}while( x == -1 && y == -1 );

closegraph(); /// inchide fereastra

În funcția repetitivă alăturată, verificăm dacă utilizatorul a apăsat click stânga. Dacă a făcut acest lucru, fereastra cu graficul se închide.

Funcția closegraph() închide fereastra de desenat.

# Scurt manual de utilizare

Utilizarea aplicației făcută de noi este foarte simplă, deci nu este nevoie de un manual complex. Totuși ar trebui menționate câteva limitări și informații, precum:

* Citirea polinomului de la tastatură prin șir este posibilă, doar dacă este respectată următoarea sintaxă:

c1x^p1 + c2x^p2 – c3x^p3 + ... + cnx + c0, unde cn -> coeficientul lui x^pn (pn reprezintă puterea la care este ridicat x)

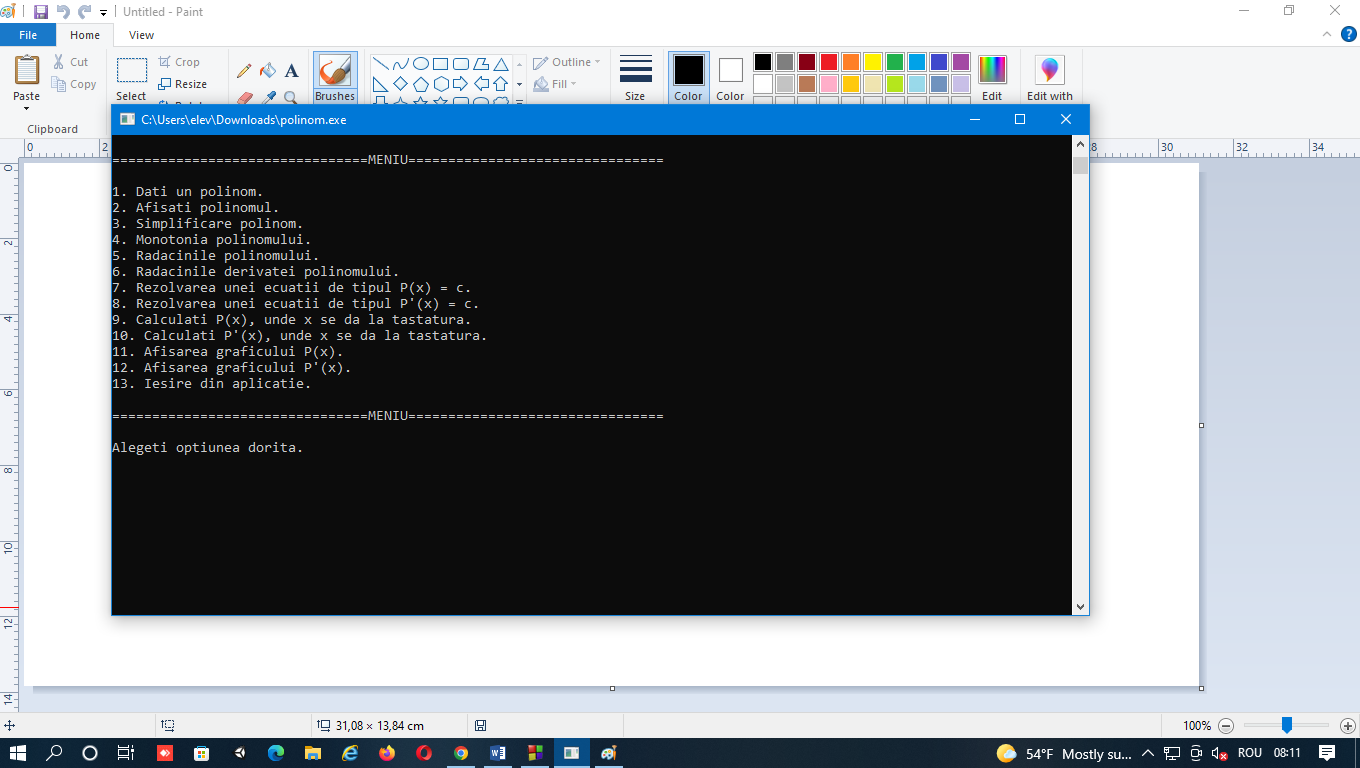
Restricții și precizări:

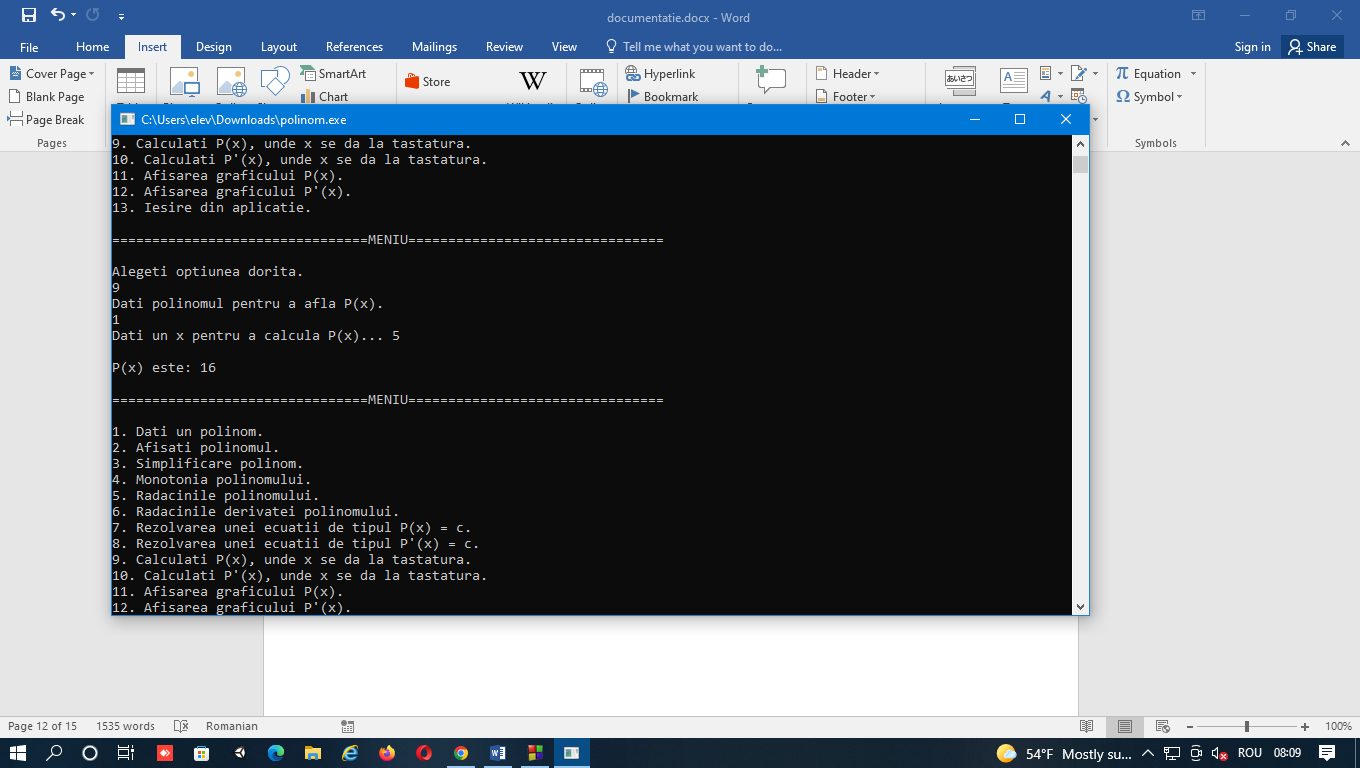
* Datele introduse prin șir pot fi doar numere intregi;
* Coeficienții pot lua valori doar intre – 100000 si 100000;
* Puterile pot lua valori doar între 0 si 99.

Pentru a utiliza aplicatia, trebuie introdus de la tastura un numar de la 1 la 13 care va initia una din comenzile afisate. Dupa alegerea unei comenzi, se va apasa „Enter” pentru a incepe executarea comenzii respective. Daca se introduce un polinom, intai trebuie introdus 1 sau 2, in functie de dorinta utilizatorul de a il introduce pe baza sirului de coeficienti sau pe baza sirului de caracter, iar dupa aceea se introduce polinomul efectiv conform instructiunilor de mai sus.

La simplificarea polinomul trebuie introdus numarul de ordine al polinomul pe care utilizatorul doreste sa il simplifice. Daca utilizatorul nu introduce corect un numar de ordine al unui polinom introdus ... atunci va trebui sa introduca un numar pana introduce unul corect. Aceeasi situatie este intalnita la monotonia polinomului, radacini, rezolvarea unei ecuatii, dar si la afisarea graficului.

Calcularea la P(x) sau P’(x) se face foarte simplu, introducand o valoare de la tastatura.





# Concluzii și posibilele extinderi

Această clasă poate să fie folosită în predarea în matematică a polinoamelor, deoarece avem aproximativ toate elementele lui.

Din acest proiect, am învățat să lucrăm mai bine cu grafica în c++ și ne-am îmbogățit experiența cu multe idei de rezolvare a bug-urilor ☺.

Posibilele extinderi:

- algoritmi mai buni de calculare a rădăcinii polinomului;

- graficul poate să aibă mai multe funcționalități (sa arătăm

mai multe grafice de polinoame în aceeași fereastră, să conțină puncte de intersecție cu axele Ox și Oy, zoom in și zoom out la grafic – ca să îl vedem în anumite puncte mai bine – );

- să fie calculat cu o precizie mai mare ecuații de tipul P(x) = 0 sau P’(x) = 0;

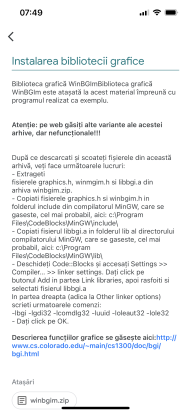
- rădăcinile polinomului să fie calculate cât mai exact (deocamdată algoritmii de rezolvare a rădăcinii calculează cu maxim 2 zecimale exacte...).

# Tips and tricks

Oferim kit de instalare a librăriei graphic.h. O să atașăm pașii de instalare.

Fișierul “winbgim.zip” poate fi luat de pe internet. În ajutorul dumneavoastră, vă oferim un link de unde puteți lua arhiva zip.

* <http://winbgim.codecutter.org/V6_0/WinBGIm_Library6_0_Nov2005.zip>



Dacă vreți ca aplicația să meargă... vă rugăm să dați niște polinoame care să fie decente.

Dacă introduceți polinoame de tipul x^900 + 2\* x ^ 42 + 500 graficul acestuia nu se va încadra în fereastră ☹.

Ca să închideți fereastra cu graficul puteți apăsa click stânga și se închide.

# Bibliografie

* <https://home.cs.colorado.edu/~main/bgi/doc/>
* <https://stackoverflow.com/questions/5029840/convert-char-to-int-in-c-and-c>
* <https://www.cplusplus.com/reference/cmath/?kw=cmath>
* <https://www.cplusplus.com/reference/istream/iostream/?kw=iostream>
* <https://www.cplusplus.com/reference/cstring/?kw=cstring>
* <https://www.cplusplus.com/reference/cctype/?kw=cctype>
* <https://www.cplusplus.com/reference/fstream/fstream/?kw=fstream>