

BIBLIOTEKA DZIAŁAŃ NA MACIERZACH

Autor: Magdalena Pastuła

Spis treści:

1. Wstęp	4
2. Analiza problemu	5
3. Rozwiązanie techniczne.....	8
Mechanizm obliczeniowy- przedstawienie macierzy w komputerze.....	8
Opracowanie interfejsu graficznego	8
Uwagi ogólne.....	9
4. Opis realizacji-opis platformy testowej	10
5. Używanie zaprojektowanego kalkulatora macierzy.....	11

Lista używanych oznaczeń:

$M_{m,n}$	Macierz o wymiarach $m \times n$
$m_{i,j}$	Wyraz macierzy M z wiersza i oraz kolumny j
E	Macierz jednostkowa
S	Macierz symetryczna
A^T	Macierz transponowana A
$\det(A)$	Wyznacznik z macierzy A
$\text{rank}(A)$	Rząd macierzy A
$\text{def}(A)$	Defekt macierzy A
$\text{tr}(A)$	Ślad macierzy A

1. Wstęp.

Dokument opisuje działanie aplikacji o nazwie kalkulator macierzy, jak również jego cechy od strony technicznej, sposób wykonania i inne, wymienione w spisie treści.

2. Analiza problemu.

Macierzą $M_{m,n}$ nazywamy obszar o m wierszach i n kolumnach. Przykładem macierzy może być:

$$M_{m,n} = \begin{bmatrix} 1 & 10 \\ 6 & 4 \end{bmatrix}$$

Ponadto, symbolem $m_{i,j}$ oznaczamy wyraz w "i-tym" wierszy oraz "j-tej" kolumnie macierzy $M_{m,n}$.

Na macierzach można dokonywać następujących operacji:

- a) porównanie macierzy

Które jest zdefiniowane następująco:

$$A_{m,n} = B_{m,n} \Leftrightarrow \bigwedge_{\substack{i=1,2,\dots,m \\ j=1,2,\dots,n}} a_{i,j} = b_{i,j}$$

Czyli, dla przykładowych macierzy $A_{3,4}$ oraz $B_{3,4}$ otrzymamy:

- b) dodawanie/odejmowanie dwóch macierzy

Działania te są zdefiniowane następująco:

$$A_{m,n} \pm B_{m,n} = C_{m,n} \Leftrightarrow \bigwedge_{\substack{i=1,2,\dots,m \\ j=1,2,\dots,n}} c_{i,j} = a_{i,j} \pm b_{i,j}$$

- c) mnożenie macierzy przez skalar

Zdefiniowane jako:

$$A_{m,n} * b = C_{m,n} \Leftrightarrow \bigwedge_{\substack{i=1,2,\dots,m \\ j=1,2,\dots,n}} c_{i,j} = a_{i,j} * b$$

- d) mnożenie macierzy przez macierz

Zdefiniowane jako:

$$A_{m,n} * B_{n,p} = C_{m,p} \Leftrightarrow \bigwedge_{\substack{i=1,2,\dots,m \\ j=1,2,\dots,p}} c_{i,j} = \sum_{k=1}^n a_{i,k} * b_{k,j}$$

e) transponowanie macierzy

Zdefiniowane jako:

$$B_{n,m} = A_{m,n}^T \Leftrightarrow \bigwedge_{\substack{i=1,2,\dots,m \\ j=1,2,\dots,n}} b_{i,j} = a_{j,i}$$

f) obliczanie wyznacznika z macierzy

Zdefiniowane rekurencyjnie jako:

$$\begin{aligned} \det(A_{1,1}) &= a_{1,1} \\ \det(A_{m,m}) &= \sum_{i=1}^m (-1)^{i+1} * a_{1,i} * M_{1,i} \end{aligned}$$

Gdzie $M_{1,i}$ jest wyznacznikiem z macierzy powstałej z wykreślenia z macierzy pierwszego wiersza oraz i-tej kolumny.

Ponadto, wyznacznik można obliczyć jedynie z macierzy kwadratowej.

g) znajdowanie dopełnienia macierzy oraz macierzy dopełnień

Zdefiniowane jako:

$$A_{i,j} = (-1)^{i+1} * M_{1,i}$$

Czyli wyznacznik z macierzy powstałej poprzez wykreślenie i-tego wiersza o j-tej kolumny.

Działanie to, analogicznie do obliczania wyznacznika macierzy, jest tylko możliwe dla macierzy kwadratowej.

h) znajdowanie rzędu macierzy

Zdefiniowane jako wymiar maksymalnego nie zerowego minora macierzy. Definicja odwołuje się również do macierzy prostokątnych. Rząd macierzy można również łatwo znaleźć poprzez eliminację Gaussa.

i) obliczanie śladu macierzy kwadratowej oraz defektu macierzy

Zdefiniowane jako:

$$tr(A_{m,m}) = \sum_{i=1}^m a_{i,i}$$

$$def(A_{m,n}) = \min(m,n) - rank(A_{m,n})$$

j) znajdowanie macierzy odwrotnej, jeśli jest to możliwe

Zdefiniowane jako:

$$B_{m,m} = A_{m,m}^{-1} \Leftrightarrow A_{m,m} * B_{m,m} = E$$

k) znajdowanie macierzy trójkątnej

Czyli przekształcanie macierzy do macierzy trójkątnej dolnej przy użyciu prostej metody eliminacji Gaussa. Operacja ta może się odnosić również dla macierzy prostokątnej. Wtedy przekształcamy ją do macierzy pseudo trójkątnej.

Jak można zauważyć, niektóre operacje takie jak porównanie dwóch macierzy są krótko mówiąc trywialne i można z łatwością wykonać je ręcznie, to operacje takie jak wyznacznik z macierzy, albo znalezienie macierzy odwrotnej zaczynają już być problematyczne, szczególnie dla macierzy o wymiarach większych niż 3x3. W związku z tym przy tworzeniu aplikacji do operacji na macierzach postanowiono skupić się na następujących operacjach oznaczonych powyżej literami d - k.

Natomiast pozostałe operacje również postanowiono zaimplementować, głównie w celach pomocniczych do innych obliczeń oraz w razie późniejszego wykorzystania kodu w innych projektach, jednakże nie można tych działań bezpośrednio w aplikacji wykonywać. Wyjątkiem jest działanie dodawania i odejmowania dwóch macierzy, których również można w aplikacji używać, a zrobiono to głównie ze względów tradycyjnych oraz w razie potrzeby obliczenia sumy lub różnicy macierzy o nieprzyjemnych wartościach.

3. Rozwiązania techniczne.

Podczas pisania całego programu starano się, aby całość była dosyć przejrzysta i łatwa do odczytania dla osoby chcącej zapoznać się z kodem. Z tego powodu podzielono jego tworzenie na dwie części - stworzenie biblioteki do działań na macierzach oraz stworzenie aplikacji okienkowej. Mimo to, w całym programie starano się trzymać takich samych lub przynajmniej podobnych oznaczeń, jak np. zmienna o nazwie `v` jest zarezerwowana dla aktualnego wiersza w macierzy podczas przechodzenia przez pętlę. Wszystkie tego typu oznaczenia są opisane na początku każdego pliku nagłówkowego, chyba, że oznaczenie między plikami się powtarza, wtedy jest opisane tylko jeden raz. Ponadto, przy ważniejszych funkcjach znajduje się krótki jej opis, a ich nazwy są tak dobrane, aby kojarzyły się z danym działaniem. Np. metoda `det()` w klasie `matrix` opisującej macierz, która oblicza wyznacznik z macierzy.

a) mechanizm obliczeniowy - przedstawienie macierzy w komputerze.

Narzędzie do działań na macierzach postanowiono zrobić w formie osobnej biblioteki, czy też osobnego pliku nagłówkowego. Powodem było wstępne założenie, że tworzona biblioteka ma służyć nie tylko w formie aplikacji, ale również w przyszłych projektach jako przedstawienie macierzy w pliku źródłowym.

Samą macierz przedstawiona jest w kodzie jako klasa zawierająca wskaźnik na tablicę dwuwymiarową z wartościami macierzy oraz dwoma zmiennymi przechowującymi jej wymiary. Zmienne te są stałe i poza momentem inicjalizowania klasy nie można ich zmienić. Natomiast wartości macierzy można zmienić stosując odpowiednie metody - na wpisanie jednej wartości w odpowiednie miejsce lub nadpisanie całej macierzy. Dostępne są również inne metody, jak wczytanie tablicy wartości, konstruktory, a także metody pozwalające obliczać wszystkie wyrażenia zdefiniowane powyżej: dodawanie, odejmowanie, mnożenie macierzy (zdefiniowane jako przeciążenia operatorów) oraz pozostałe, jako zwykłe metody. Ponadto klasa zawiera pewne metody pomocnicze do określonych obliczeń, jak np. `move_rows()` pomocny przy obliczaniu wyznacznika. Jednakże są one metodami prywatnymi, a zatem niemożliwe do wywołania poza klasą.

Część problemów, ze względu na większą wygodę implementowania jak i późniejszego czytania kodu, została przedstawiona rekurencyjnie - jak np. obliczanie wyznacznika z macierzy. Z tego względu zdecydowano się również na ograniczenie rozmiaru macierzy ze względu na ograniczone zasoby komputera, tudzież pojemność stosu, którego pojemność jest ograniczona. Zatem dla zbyt dużych wymiarów mogłoby nastąpić jego przepełnienie, a tym samym zakończenie programu niepowodzeniem.

b) opracowanie interfejsu graficznego

Ponieważ program nie potrzebuje tak naprawdę żadnych efektów graficznych, postanowiono, że będzie on wykonany w formie aplikacji okienkowej. Ponadto, szczególną rolę grała tutaj potrzeba wpisania wartości macierzy w miarę przystępny sposób. Jednym sposobem na zaradzenie temu okazało się być umieszczenie pola tekstowego, do którego można by te wartości wpisać. Oczywiście takie rozwiązanie wiąże się z pewnymi problemami, jak np. wpisanie znaku innego niż cyfra, jednakże jest również najbardziej wygodne i zapewnia jak największą swobodę we wpisywaniu liczb. Natomiast podczas

tworzenia programu zadbane, aby przy każdym błędnym wpisaniu wartości użytkownik był o tym informowany.

Podobnie jak przypadku tworzenia samej biblioteki do działań na macierzach, tak podobnie podczas tworzenia aplikacji okienkowej starano się, aby w miarę możliwości jak najwięcej funkcji oraz stałych znalazło się w pliku nagłówkowym. Miało to na celu zwiększenie czytelności i uporządkowanie kodu. Niestety, nie udało się tak zrobić ze wszystkimi funkcjami. Niektóre musiały pozostać w pliku głównym, ze względu na potrzebę dostępu do zmiennych globalnych, takich jak uchwyt do pól tekstowych lub list rozwijalnych.

Zmienne te zostały zainicjalizowane jako globalne, aby ułatwić do nich dostęp chociażby z funkcji sterującej oknem oraz aby znacząco zmniejszyć liczbę argumentów przekazywanych do funkcji tego typu.

Praktycznie wszystkie części okna, które wymagają późniejszego rozróżnienia posiadają numer identyfikacyjny, w celu ułatwienia tego zadania. Numery te są zmiennymi predefiniowanymi procesora i są definiowane w pliku nagłówkowym.

c) uwagi ogólne

Dodatkowo, poza plikami nagłówkowymi dotyczącymi biblioteki do macierzy oraz aplikacji okienkowej program posiada jeszcze jeden, łączący te dwa. Są w nim umieszczone funkcje pełniące rolę pośrednią między jednym plikiem a drugim: funkcje takie jak przekształcanie macierzy pobranej w formie tekstu na macierz w formie liczbowej i inne.

4. Opis realizacji - opis platformy testowej.

Do zaimplementowania całej aplikacji wykorzystano język C++ w jednej z najnowszych wersji, czyli C++17. Używany IDE był Visual Studio 2017 z racji posiadania wymienionego standardu języka C++. Ponadto, Visual jest wygodnym środowiskiem, które np. pozwala na kolorowanie składni lub znalezienie wszystkich odniesień w całym projekcie do danej zmiennej czy funkcji.

Natomiast do stworzenia samej aplikacji okienkowej wykorzystano bibliotekę WinApi, z racji możliwości wykorzystania pola tekstowego, którego istota przy tworzeniu została opisana wcześniej, oraz problemów w próbach zainstalowania innych bibliotek. Ponieważ jednak WinApi jest biblioteką do pisania aplikacji okienkowych specjalnie pod system operacyjny Windows, zatem program może się nawet nie uruchomić na innych platformach.

5. Używanie zaprojektowanego kalkulatora macierzy.

Po utworzeniu aplikacji pojawia się okno podzielone na cztery części: Macierz A, czyli pole do zadeklarowania rozmiarów macierzy oraz wpisania jej wartości, analogiczna część Macierz B z dodatkowym przyciskiem informującym program, aby tą macierz również wziąć pod uwagę w obliczeniach (dotyczy operacji dwuargumentowych, jak dodawanie macierzy). Do tego mamy również pole podpisane jako "Wybranie działania", czyli rozwijalna lista z dostępnymi działaniami oraz przycisk o nazwie "Oblicz", oraz czwarte pole podpisane jako "Wynik", czyli miejsce zarezerwowane na wypisanie ewentualnego wyniku.

Aby móc skorzystać z kalkulatora, najpierw należy wybrać rozmiary macierzy z rozwijalnej listy i wpisać wartości macierzy do pola tekstowego, w razie konieczności postąpić podobnie w części nazwanej "Macierz B" zaznaczając przycisk "Uwzględnij w obliczeniach". Co ważne, przy wpisywaniu wartości macierzy należy kolejne kolumny oddzielić spacjami, natomiast kolejne wiersze przyciskiem ENTER. W innym przypadku wartości nie zostaną poprawnie pobrane i zostanie zgłoszony błąd. Dotyczy to również sytuacji, gdy kolejne kolumny oddzielimy tabulatorem, a nie spacjami.

Następnie należy wybrać działanie z rozwijalnej listy i nacisnąć przycisk "Oblicz". W części nazwanej "Wynik" powinien pojawić się rezultat obliczeń. W przypadku jakichkolwiek niezgodności we wprowadzonych danych, np. wpisaniu jednej macierzy, gdy do obliczenia są konieczne dwie lub wpisaniu macierzy o innych wymiarach niż podane, po naciśnięciu przycisku "Oblicz" wyskoczy okienko z odpowiednią informacją. Sytuacja taka ma również miejsce, gdy w pole tekstowe zostanie wpisana litera bądź inny znak nie będący cyfrą. Obejmuje to także przypadki niezastosowania się do reguł wpisania wartości macierzy określonych w poprzednim akapicie.

Po wyskoczeniu okna i potwierdzenia otrzymania informacji program zaprzestanie obliczeń - a zatem w polu "Wynik" nie zostanie nic wypisane. Należy wtedy poprawić dane i spróbować jeszcze raz. Wyjątkiem jest sytuacja, gdy do macierzy B zostaną wpisane jakieś wartości, natomiast nie zostanie zaznaczony przycisk "Uwzględnij w obliczeniach". W takim przypadku również wyskoczy okienko dialogowe, jednakże w zależności od decyzji użytkownika program te dane uwzględni lub nie. Po zdecydowaniu program będzie kontynuował działania i obliczenia, w ramach możliwości.

Jedynym wyjątkiem od zgłaszania problemów jest sytuacja, gdy zostaną wpisane dwie macierze, natomiast do obliczeń potrzebna jest tylko jedna. W takim przypadku program zignoruje macierz B i wykona odpowiednie obliczenia na macierzy A.

Po wyświetleniu wyniku program kontynuuje swą pracę. W razie potrzeby można ponownie uzupełnić macierze i wykonać inne działanie lub zamknąć program poprzez zamknięcie okna.