1. Macierze.

Podstawowym typem zmiennej Matlaba jest macierz dwuwymiarowa. Szczególnymi odmianami macierzy są:

- **skalar** - macierz o rozmiarach `1x1`,
- **wektor wierszowy** - macierz o jednym wierszu,
- **wektor kolumnowy** - macierz o jednej kolumnie.

2. Definiowanie macierzy.

Definiowanie macierzy wymaga uwzględnienia następujących reguł:

- elementy w wierszu macierzy muszą być oddzielane spacją lub przecinkiem,
- średnik lub znak nowego wiersza kończy wiersz macierzy i powoduje przejście do następnego,
- cała lista elementów musi być ujęta w nawiasy kwadratowe.

Najprostszą metodą definiowania macierzy jest podanie jej wszystkich elementów wiersz po wierszu. W definicji liczba elementów w każdym wierszu musi być jednakowa, w przeciwnym wypadku zostanie wyświetlony komunikat o błędzie.

3. Generowanie i weryfikacja wektorów oraz tablic.

Podczas generowania wektorów stosowana jest często notacja dwukropkowa. Jej postać jest następująca: `j:i:k` określa wektor `[j,j+i,j+2i,...,k]`, przy czym `i` jest krokiem, natomiast `j`, `k` są odpowiednio pierwszym i ostatnim elementem wektora.

Element macierzy znajdujący się w wierszu o indeksie `i` oraz kolumnie o indeksie `j` jest określony jako `A(i,j)`. Elementem takim można się posługiwać jak każdą zmienną - zwracać jego wartość, i przypisywać nową. Do elementów macierzy można się też odwoływać przy użyciu tylko jednego indeksu, np. `A(k)`. Jeśli zmienna `A` byłaby wektorem wierszowym lub kolumnowym, odwołanie takie oznaczałoby oczywiście `k`-ty element wektora. Natomiast, gdy `A` będzie macierzą dwuwymiarową, odwołanie takie zostanie zinterpretowane jako odwołanie do elementu macierzy uformowanej z kolejnych kolumn macierzy oryginalnej, umieszczonych jedna pod drugą.

Niżej przedstawiono sposób wyboru odpowiednich wierszy, kolumn oraz elementów macierzy:

`A(:,j)`	- wypisanie `j`-tej kolumny macierzy `A`,
`A(:,j:k)`	- wypisywanie kolumn od `j` do `k`,
`A(i,:)`	- wypisanie `i`-tego wiersza macierzy `A`,
`A(:)`	- wypisanie wszystkich elementów macierzy w jednej kolumnie; powstaje wektor zawierający elementy macierzy wypisywane kolumnami - od pierwszej do ostatniej,
`A(j:k)`	- wypisanie w jednym wierszu, elementów macierzy `A` począwszy od elementu o indeksie `j` aż do indeksu `k`.

Kolumny lub wiersze można też wskazać przez podanie ich numerów w nawiasach kwadratowych. Tak więc zapis `A(:,[2,3])` jest równoważny zapisowi `A(:,2:3)`.

Ćwiczenie 1.

Wykonaj polecenia pokazujące przykłady zastosowania notacji dwukropkowej.

```
>> d = 1:5
>> n = 9; 1:n % wektor, np. dla instrukcji iteracyjnych
>> y = 0:0.5:2
>> z = 5:-1:0
```

Ćwiczenie 2.

Wykonaj polecenia z użyciem notacji dwukropkowej w odniesieniu do macierzy.

```
>> M = [1:10; 1:2:20]
>> A =[1 2 3; 4 5 6; 7 8 9] % zdefiniowanie macierzy A
>> A(:,3) % wypisywanie 3-ciej kolumny macierzy A
```

```
>> A(:,2:3) % wypisywanie 2-giej i 3-ciej kolumny macierzy A
>> A(:,2:-1:1) % wypisanie 2-giej, a potem 1-szej kolumny
>> A(:,2:1) % taki zapis daje wektor pusty
>> A(2,:) % wypisanie 2-go wiersza macierzy A
>> A(2:3,:) % wypisanie 2-go i 3-ciego wiersza
>> A(:) % wszystkie elementy macierzy A jako wektor
>> A(4:8) % elementy macierzy od 4-go do 8-go
```

Wprowadzenie słowa kluczowego `end` umożliwia edycję ostatniego wiersza lub ostatniej kolumny macierzy.

```
>> A(end,:) % wypisanie ostatniego wiersza macierzy A
>> sum(A(:,end)) % suma elementów ostatniej kolumny macierzy A
```

Usuwanie wybranych wierszy i kolumn macierzy oraz pewnych sekwencji jej elementów wykonuje się z zastosowaniem pary nawiasów prostokątnych `[]`.

```
>> X = A; X(2,:) = [] % usunięcie 2-go wiersza macierzy X
>> Y = A; Y(1:2:8) = [] % usunięcie sekwencji elementów Y
>> B = [.1 .2 .3 .4; .5 .6 .7 .8; .9 1 1.1 1.2] % zdefiniowanie macierzy B
>> A(:,[2,3]) = B(:,3:4) % wyjaśnij działanie tego polecenia
>> A(3,3) = A(1,1) + A(2,1)
```

4. Wyświetlanie macierzy i ich rozmiarów.

W tabeli 6 zestawione zostały funkcje wyświetlające zawartość i rozmiar macierzy.

Tabela 6. Funkcje wyświetlające macierze i ich rozmiary

Funkcja	Opis
`disp(A)`	wyświetla zawartość macierzy `A` w oknie poleceń
`size(A)`	wyświetla rozmiar dwuwymiarowej macierzy `A` (liczbę wierszy i kolumn) w postaci dwuelementowego wektora wierszowego `[liczba wierszy liczba kolumn]`
`[n m]=size(A)`	przypisuje zmiennej `n` liczbę wierszy, a zmiennej `m` - liczbę kolumn macierzy `A`
`n = size(A,1)`	przypisuje zmiennej `n` liczbę wierszy macierzy `A`
`m = size(A,2)`	przypisuje zmiennej `m` liczbę kolumn macierzy `A`
`length(x)`	zwraca długość wektora `x` lub dłuższy z wymiarów macierzy

Ćwiczenie 3.

Wykonaj polecenia demonstrujące uzyskiwanie informacji o wymiarach macierzy.

- Zdefiniuj macierz `D`:
```
>> D = [2 0 1; 1 5 0]
```
- Wyświetl rozmiar macierzy:
```
>> size(D)
```
- Przypisz zmiennej `ile_wierszy` liczbę wierszy, a zmiennej `ile_kolumn` liczbę kolumn macierzy `D`:
```
>> [ile_wierszy ile_kolumn] = size(D)
```
lub
```
>> ile_wierszy = size(D,1), ile_kolumn = size(D,2)
```
- Utwórz macierz `D1` o rozmiarze macierzy `D`, składającą się z samych jedynek (użyta zostanie funkcja `ones`, opisana w tabeli 8):

```
>> D1 = ones(size(D))
```

5. Arytmetyka macierzowa i tablicowa.

W szczególnym przypadku, macierz można potraktować jako zwykłą prostokątną tablicę liczb. Operacje na macierzach dokonywane są zgodnie z algebrą macierzy, natomiast operacje na tablicach wykonywane są element po elemencie. Matlab dostarcza w tym celu odrębnych arytmetycznych operatorów macierzowych i tablicowych.

Niech $A = \begin{bmatrix} 1 & -1 \\ -2 & 3 \end{bmatrix}$, $B = \begin{bmatrix} 1 & 1 \\ 0 & -2 \end{bmatrix}$.

W tabeli 7 zestawiono wyniki operacji dla danych **A** i **B**, wykorzystując operatory macierzowe i tablicowe. Operatory tablicowe poprzedzone są zawsze kropką, np.:

- **mnożenie macierzowe** - gwiazdka (*****),
- **mnożenie tablicowe** - kropka i gwiazdka (**.***).

Tabela 7. Efekty działania operatorów macierzowych i tablicowych

Operator arytmetyczny	Operator macierzowy	Operator tablicowy
dodawanie	**A+B**= $\begin{bmatrix} 2 & 0 \\ -2 & 1 \end{bmatrix}$	tak jak macierzowy
odejmowanie	**A-B**= $\begin{bmatrix} 0 & -2 \\ -2 & 5 \end{bmatrix}$	tak jak macierzowy
mnożenie	**A*B**= $\begin{bmatrix} 1 & 3 \\ -2 & -8 \end{bmatrix}$ **B*A**= $\begin{bmatrix} -1 & 2 \\ 4 & -6 \end{bmatrix}$ (brak przemienności mnożenia)	**A.*B=B.*A**= $\begin{bmatrix} 1 & -1 \\ 0 & -6 \end{bmatrix}$
dzielenie	**C1=B/A**= $\begin{bmatrix} 5 & 2 \\ -4 & -2 \end{bmatrix}$ - dzielenie prawostronne (**B=C1*A**) **C2=B\A**= $\begin{bmatrix} 0 & 0.5 \\ 1 & -1.5 \end{bmatrix}$ - dzielenie lewostronne (**B=A*C2**)	**B./A=A.\B**= $\begin{bmatrix} 1 & -1 \\ 0 & -\frac{2}{3} \end{bmatrix}$
potęgowanie	**A^2=A*A**= $\begin{bmatrix} 3 & -4 \\ -8 & 11 \end{bmatrix}$	**A.^2**= $\begin{bmatrix} 1 & 1 \\ 4 & 9 \end{bmatrix}$
transpozycja (zamiana wierszy na kolumny)	**A'**= $\begin{bmatrix} 1 & -2 \\ -1 & 3 \end{bmatrix}$	tak jak macierzowy

Podczas mnożenia macierzowego **A*B** należy dopilnować, aby liczba wierszy macierzy **A** była <u>równa</u> liczbie kolumn macierzy **B**. Ponadto, w rachunku macierzowym nie jest spełnione prawo przemienności mnożenia.

Występuje zarówno prawostronny (**/**), jak i lewostronny (****) operator dzielenia macierzowego. Z operatorów tych należy korzystać ostrożnie, gdyż otrzymane wyniki są zupełnie różne. Problemu tego nie ma w przypadku operacji na tablicach.

Transpozycja macierzy o składnikach rzeczywistych daje takie same wyniki niezależnie od tego, czy wykorzystywany jest operator macierzowy, czy tablicowy. Różnice pojawiają się, gdy składniki macierzy są zespolone.

Ćwiczenie 4.

Wykonaj polecenia demonstrujące operacje macierzowe i tablicowe.

- Zdefiniuj macierze **A** i **B**:

```
>> A = [1 0; 3 2];
>> B = [1 2; 3 4];
```

- Oblicz iloczyny macierzy **A** i **B** macierzowo i tablicowo:

```
>> A*B, B*A % mnożenie macierzowe
>> A.*B, B.*A % mnożenie tablicowe
```

- Oblicz A^3 macierzowo i tablicowo:

```
>> A^3 % mnożenie macierzy A*A*A
>> A.^3 % potęgowanie pojedynczych elementów Aij
```

- Oblicz iloczyn $(A\cdot B)^{-1}\cdot(A\cdot B)$:

```
>> G = (A*B).^(-1)*(A*B)
```

- Wyznacz macierz $C=(A+B^T)/2$, gdzie B^T - macierz transponowana

```
>> C = (A+B')./2
```

Ćwiczenie 5.

Wykonaj polecenia demonstrujące wybrane operacje na wektorze i macierzy.

- Zdefiniuj wektor wierszowy $x=\begin{bmatrix}1 & 4\end{bmatrix}$ i macierz $A=\begin{bmatrix}4 & 1\\ 7 & 2\end{bmatrix}$:

```
>> x = [1 4], A = [4 1; 7 2];
```

- Wykonaj mnożenie $A\cdot x$:

```
>> A*x
```

Nie można zapisać iloczynu w ten sposób, ponieważ liczba kolumn macierzy **A** nie jest równa liczbie wierszy wektora **x**. W takim przypadku pojawi się komunikat o błędzie:

```
??? Error using ==> mtimes
Inner matrix dimensions must agree.
```

- Prawidłowy zapis iloczynu:

```
>> A*x'
```

W powyższym zapisie `x'` jest transpozycją wektora **x**, czyli wektorem kolumnowym $\begin{bmatrix}1\\ 4\end{bmatrix}$.

Ćwiczenie 6.

Przyjmując wektor $x=\begin{bmatrix}0 & \frac{\pi}{2} & \pi & \frac{3\pi}{2} & 2\pi\end{bmatrix}$, oblicz wartości funkcji $y=2x\sin(1+x^2)$.

```
>> x = 0:pi/2:2*pi;
>> y = 2*x.*sin(1+x.^2)
```

Operacje matematyczne zostały wykonane kolejno dla każdego elementu wektora **x**. Otrzymany wektor **y** ma tyle samo elementów, co wektor **x**.

Mnożenie i dzielenie prawostronne wektora przez liczbę (lub liczby przez liczbę) daje takie same wyniki niezależnie od tego, czy użyto operatora macierzowego, czy tablicowego (w ćwiczeniu 6: `pi/2`, `2*x`). Różnice pojawiają się przy mnożeniu i dzieleniu prawostronnym wektora przez wektor, np. wyrażenie x^2 może być poprawnie zapisane tylko jako `x.*x` (w przypadku zapisu `x*x` zostanie zwrócony błąd).

6. Macierze pełne.

Macierz pełna jest rozumiana jako dwuwymiarowa tablica, na elementach której wykonuje się operacje algebry liniowej. Jak wspomniano wcześniej, w pamięci wewnętrznej przechowuje się wartość każdego jej elementu. Zestawienie wybranych funkcji stosowanych do wykonywania operacji na macierzach podano w tabeli 8.

Tabela 8. Funkcje stosowane do wykonywania operacji na macierzach

Nazwa	Opis funkcji
Funkcje przeznaczone do konstruowania macierzy	
`magie`	kwadrat magiczny
`handamard`	macierz Hadamarda
`hankel`	macierz Hankela
`hilb`	macierz Hilberta
`ones`	macierz o elementach równych jeden
`pascal`	macierz Pascala
`company`	macierz stowarzyszona wielomianu
`toeplitz`	macierz Toeplitza
`vander`	macierz Vandermonde'a
`zeros`	macierz z elementami równymi zero
`eye`	macierz z jedynkami na przekątnej
`gallery`	macierze testowe
`randn`	macierz losowa o rozkładzie normalnym
`rand`	macierz losowa o rozkładzie równomiernym
`meshgrid`	tablica dla wykresów trójwymiarowych
`logspace`	wektor o wartościach rozłożonych logarytmicznie
`linspace`	wektor o wartościach rozłożonych równomiernie
Rezultat zastosowania funkcji	
`diag`	macierz diagonalna lub wektor elementów na przekątnej
`triu`	macierz trójkątna z elementami nad główną przekątną
`tril`	macierz trójkątna z elementami pod główną przekątną
`rot90`	obrót macierzy o 90°
`flipud`	zamiana kolejności wierszy macierzy góra/dół
`fliplr`	zamiana kolejności wierszy macierzy lewo/prawo
`repmat`	powielenie macierzy w pionie i poziomie
`reshape`	zmiana wymiaru macierzy
Elementarne funkcje macierzowe	
`expm`	macierzowa funkcja wykładnicza
`logm`	macierzowa funkcja logarytmiczna
`sqrtm`	macierzowa funkcja pierwiastkowa
Funkcje analizy macierzowej	
`inv`	macierz odwrotna
`norm`	normy wektora i macierzy
`lu`	rozkład na macierze trójkątne LU
`ploy`	współczynniki wielomianu charakterystycznego
`det`	wyznacznik macierzy

Ćwiczenie 7.

Dla danych macierzy **A** i **B** z ćwiczenia nr 4 wyznacz macierz $(A \cdot B)^{-1}$.

```
>> D = inv(A*B) % macierz odwrotna do iloczynu macierzy A·B
```
lub
```
>> D = (A*B)^(-1)
```

Ćwiczenie 8.

Zdefiniuj macierz $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, a następnie wykonaj poniższe polecenia.

- Powiel macierz **A** dwa razy w pionie i trzy razy w poziomie:
```
>> B = repmat(A,2,3)
```

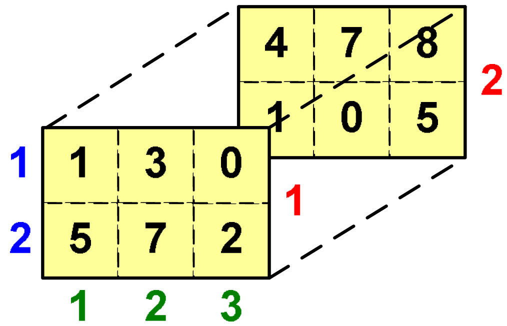
- Zmień rozmiar macierzy **B** utworzonej w poprzednim punkcie na 2 wiersze i 12 kolumn:
```
>> C = reshape(B,2,12)
```

7. Tablice wielowymiarowe.

Tablic wielowymiarowych używa się do przechowywania zestawów tablic o jednakowych wymiarach, które również mogą być wielowymiarowe. Elementy tablic wielowymiarowych wskazuje się przez podanie nazwy tablicy i indeksów w nawiasie okrągłym, na przykład `AAA(2,3,5)`. Przyjmuje się, że indeksy oznaczają:

- pierwszy indeks - wiersz macierzy (wymiar 1),
- drugi indeks - kolumnę macierzy (wymiar 2),
- trzeci indeks - stronę macierzy (wymiar 3 i następne).

Na poniższym rysunku widoczna jest macierz trójwymiarowa o rozmiarze `2x3x2` (2 wiersze i 3 kolumny na każdej stronie, 2 strony).



Ćwiczenie 9.

Zdefiniuj macierz o rozmiarze `2x3x2` przedstawioną na powyższym rysunku, a następnie odwołaj się do wybranego elementu tej macierzy.

- Macierz wielowymiarową taką definiuje się stronami:
```
>> D(:,:,1) = [1 3 0; 5 7 2] % strona nr 1
>> D(:,:,2) = [4 7 8; 1 0 5] % strona nr 2
```

- Odwołanie do elementu np. w pierwszym wierszu, drugiej kolumnie, na stronie drugiej ma postać
```
>> D(1,2,2)
```

Na tablicach wielowymiarowych wykonuje się pojedyncze operacje za pomocą funkcji i operatorów, które działają na pojedynczych elementach tablic. Własności takich tablic można uzyskać poleceniem **nddemo**.

Tablice wielowymiarowe umożliwiają organizowanie uporządkowanych zestawów danych dwuwymiarowych w postaci kolejnych stron tablicy trójwymiarowej.

Tabela 9. Funkcje operujące na strukturze macierzy

Nazwa	Opis funkcji
`ipermute`	odwraca efekt permutacji
`ndims`	podaje liczbę wymiarów, pomija wymiary nieistotne
`cat`	powiększa tablice przez doklejenie innych tablic
`permute`	przestawia wiersze, kolumny i warstwy tablicy
`shiftdim`	przesuwa indeksy tablicy (+/- w lewo lub prawo)
`squeeze`	usuwa nieistotne indeksy tablicy

Ćwiczenie 10.

Wykonaj polecenia demonstrujące operacje na tablicach wielowymiarowych. Jako przykład, utworzona zostanie tablica trójwymiarowa o wymiarze **(5x4x2)**, której wybrane elementy zostaną wypełnione liczbami.

```
>> F(5,4,1) = 541; F(4,2,1) = 421;... % kontynuacja polecenia w następnym wierszu
   F(:,:,2) = 222; F(1,4,2) = 142; F(4,4,2) = 442
```

Tablicę tego typu można powiększyć, dopisując dodatkowy zerowy element w wymaganym punkcie przestrzeni n-wymiarowej, na przykład:

```
>> F(5,5,5,2) = 0;
```

Na podstawie dwóch zdefiniowanych tablic **A** i **B** utworzona zostanie trzecia tablica trójwymiarowa **C**, w wyniku warstwowego ułożenia tablic **A** i **B**.

```
>> A = [1,2,3; 21,22,23];
>> B = A + 100;
>> C = cat(3,A,B)
```

Możliwe jest także rozszerzenie istniejącej tablicy **A** do trzech wymiarów, na przykład:

```
>> A(:,:,2) = B
```

Dzięki takiej operacji uzyskuje się dwie takie same tablice **A** i **C**.

Ćwiczenie 11.

Wykonaj polecenia z użyciem operatorów relacji oraz wyrażeń logicznych w odniesieniu do dwóch macierzy.

```
>> A = [-1 2 0; 4 0 6; 7 8 -9]
>> B = [1 2 3; 1 2 3; 1 2 3]
>> A >= B
>> A & B % iloczyn logiczny macierzy A i B
>> ~A % negacja macierzy A
>> x = 5; x >= A
```

8. Zadania.

Zadanie 1.

Utwórz macierze **A** i **B** oraz wektor **f** określone jako:

$$A = \begin{bmatrix} 2 & -7 \\ 5 & 4 \end{bmatrix},\ B = \begin{bmatrix} 6 & 1 \\ 4 & -3 \end{bmatrix},\ f = \begin{bmatrix} 4 & 1 \end{bmatrix}.$$

Zadanie 2.

Wykonaj polecenia:

a) wyświetl rozmiar macierzy **A** oraz wektora **f** z zadania nr 1,
b) oblicz transpozycję macierzy **B**,
c) oblicz wyrażenie **(A+B)**^{2}**+2(A-B)**,
d) utwórz macierz **C=[[A] [B]]** i wektor **h=[f f]**,
d) oblicz iloczyn **C·h**.

Zadanie 3.

Wykonaj polecenia:

a) utwórz 24-elementowy wektor **x=[1 2 ... 24]**,
b) za pomocą funkcji **reshape** utwórz z wektora **x** macierz **Y** o postaci

$$Y = \begin{bmatrix} 1 & 7 & 13 & 19 \\ 2 & 8 & 14 & 20 \\ \dots & \dots & \dots & \dots \\ 6 & 12 & 18 & 24 \end{bmatrix}.$$

Zadanie 4.

Utwórz macierze o rozmiarze **3x4**:

a) o wszystkich elementach równych **1**,
b) o wszystkich elementach równych **0**,
c) wypełnioną liczbami pseudolosowymi.

Zadanie 5.

Wygeneruj dane do wykresu funkcji $y = x \cdot \sin(x)$ dla $x = 0,\ 0{,}1,\ 0{,}2,\ \dots,\ 10{,}0$ i uzyskane wyniki zapisz w pliku tekstowym na dysku.

Zadanie 6.

Rozwiąż niżej podany układ równań.

$$\begin{cases} 3x_1 + 2x_2 - x_3 = 5 \\ x_1 + x_2 + x_3 = 11 \\ 2x_1 - x_2 + 4x_3 = 8 \end{cases}$$