

Technologie informacyjne

Wprowadzenie do Matlaba

Rafał Kabaciński

4 listopada 2019

Spis treści

1	Czym jest Matlab	1
2	Podstawy Matlaba	1
2.1	Okno programu	1
3	Podstawowe komendy	4
3.1	Praca ze zmiennymi	4
3.2	Podstawowe działania	5
3.3	Definiowanie macierzy	5
3.4	Podstawowe funkcje	6
4	Skrypty	6
5	Wykresy	7

1 Czym jest Matlab

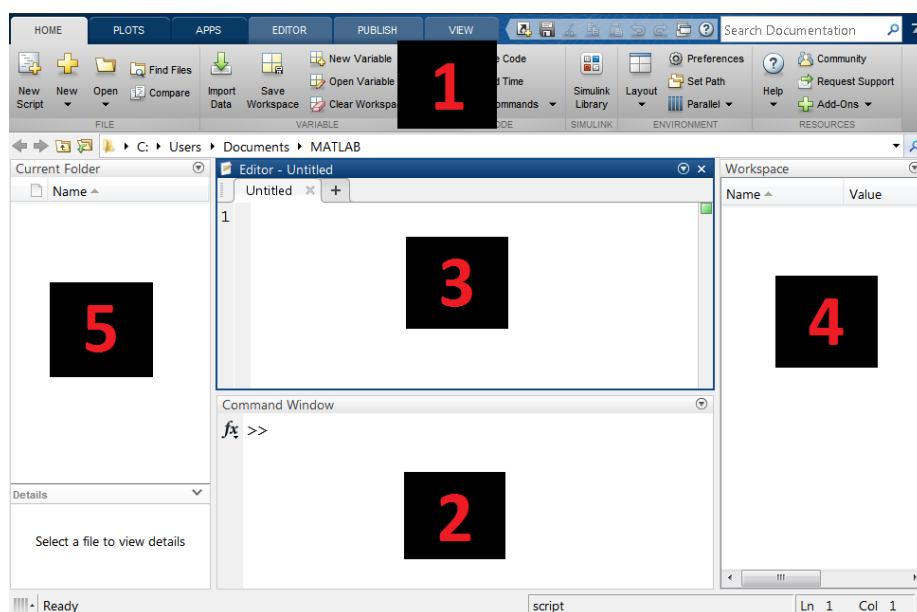
Matlab to zintegrowane środowisko do obliczeń matematycznych, inżynierskich i prowadzenia symulacji oraz powiązanego z nim języka programowania. Jego nazwa wywodzi się od MATrix LABoratory, czyli laboratorium macierzy. Obecnie jednak jego możliwości wykraczają daleko poza obliczenia macierzowe, a sam program stosowany jest w wielu różnych dziedzinach. Poza podstawowym środowiskiem dostępne są liczne specjalizowane biblioteki narzędzi nazywane *toolboxami*. Pośród alternatyw dla Matlaba wyróżnia się otwarto-źródłowy pakiet Octave, którego jednym z celów jest kompatybilność z językiem Matlab.

Ponieważ pierwotnym przeznaczeniem środowiska było prowadzenie interaktywnych obliczeń, Matlab jest językiem interpretowanym a nie kompilowanym w przeciwieństwie do na przykład języków C, C++.

2 Podstawy Matlaba

2.1 Okno programu

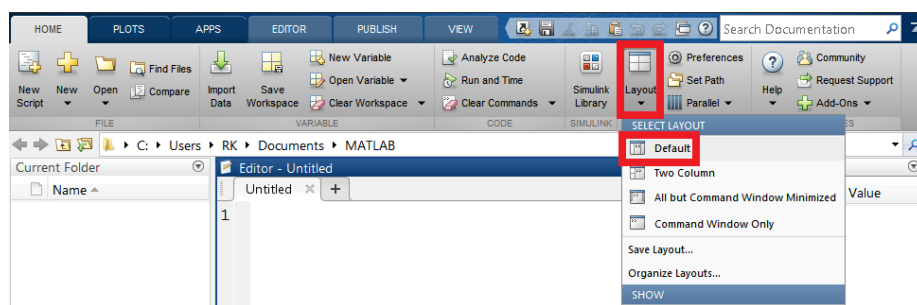
Główne okno programu Matlab zaprezentowano na ilustracji 1.



Ilustracja 1: Okno programu Matlab; 1 – wstążki narzędzi, 2 – linia komend, 3 – edytor skryptów, 4 – przestrzeń robocza, 5 – pliki w bieżącym folderze

Narzędzia pogrupowane są w tak zwane wstążki, znajdujące się u góry okna (Ilustracja 1–1). Linia komend (Ilustracja 1–2) znajdująca się u dołu okna służy do prowadzenia interaktywnych obliczeń. Bardziej rozbudowane obliczenia umieścić można w plikach skryptów. Edytor takich plików znajduje się pośrodku (Ilustracja 1–3). Po prawej stronie znajduje się przestrzeń robocza ((Ilustracja 1–4), a po lewej podgląd plików w bieżącym folderze (Ilustracja 1–5).

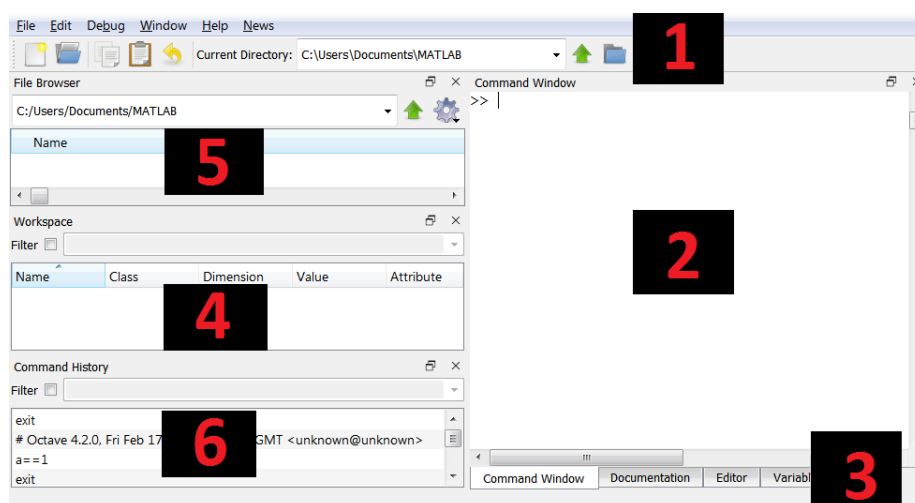
Jeżeli wygląd okna został zmieniony względem domyślnego, można go przywrócić. Służy do tego narzędzie *Layout* znajdujące się po środku wstążki *Home* (Ilustracja 2).



Ilustracja 2: Przywrócenie domyślnej konfiguracji okna Matlaba

Główne okno programu Octave ma zbliżoną konfigurację, zaprezentowane jest na ilustracji 3.

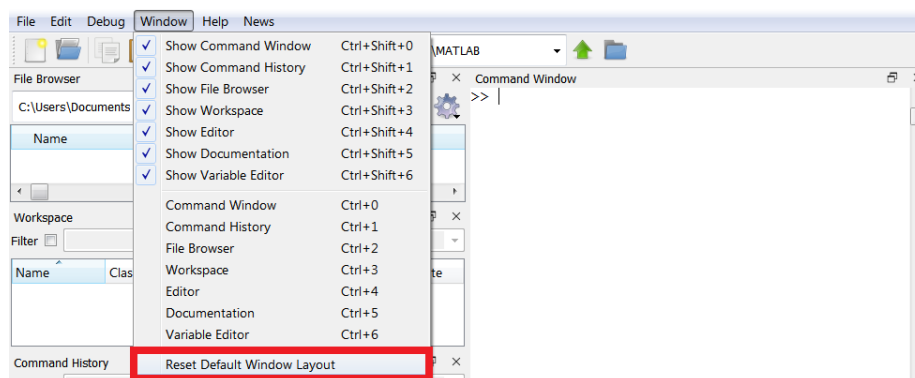
Podstawowe narzędzia dostępne są u góry okna (Ilustracja 3–1). Linia komend (Ilustracja 3–2) znajduje się po prawej. Może ona być jednak przełączona



Ilustracja 3: Okno programu Octave; 1 – narzędzia, 2 – linia komend, 3 – zakładki środkowego sub-okna, 4 – przestrzeń robocza, 5 – pliki w bieżącym folderze, 6 – historia komend

między innymi na edytor skryptów za pomocą zakładek dostępnych u dołu okna (Ilustracja 3–3). Po lewej stronie, po środku znajduje się przestrzeń robocza ((Ilustracja 1–4). Nad nią z kolei umieszczono podgląd plików w bieżącym folderze (Ilustracja 1–5). Pod przestrzenią roboczą znajduje się sub-okno z historią komend.

Jeżeli wygląd okna został zmieniony względem domyślnego, można go przywrócić. Służy do tego ostatnia opcja z menu *Window* (Ilustracja 4).



Ilustracja 4: Przywrócenie domyślnej konfiguracji okna Octave

3 Podstawowe komendy

Zadanie: wpisz:
`a==1`
w linii komend i wciśnij enter.
Przeczytaj informację o błędzie

Linia komend służy do prowadzenia interaktywnych obliczeń. Kolejne komendy potwierdza się za pomocą klawisza *enter*. W przypadku błędu zamiast wyniku pojawia się informacja o błędzie zapisana czerwoną czcionką (czarną w Octave). Informacje te przeważnie szczegółowo informują jaki błąd pojawił się w komendzie, czasem sugerując nawet poprawki.

3.1 Praca ze zmiennymi

Do definiowania macierzy i wektorów służą nawiasy kwadratowe. Kolejne elementy w wierszu rozdziela się spacją bądź przecinkiem. Wiersze separuje się za pomocą średnika:

```
a=[1 2]
b=[1 2; 3 4]
```

Uwaga: Znak `=` w większości języków programowania nie jest operatorem relacji równości jak w matematyce, tylko operatorem przypisania wartości po prawej do zmiennej po lewej jego stronie!

Zadanie: Zdefiniuj zmienne `a` i `b` jak w przykładzie

Zmienne zdefiniowane w trakcie sesji pojawiają się w przestrzeni roboczej. Można w niej sprawdzić nazwy, wymiary czy wartości (dla mniejszych macierzy). Typ zmiennym nadawany jest automatycznie przez środowisko zależnie od tego co zostanie do nich przypisane.

Kolejne polecenia wpisane w linii komend zapisywane są do historii. Można się do nich odwołać za pomocą strzałek góra/dół na klawiaturze. W Matlabie wciśnięcie strzałki w górę wywołuje sub-okno historii, w Octave jest ono domyślnie widoczne cały czas po lewej stronie. Wywołane z historii komendy można edytować.

Ciągi znakowe definiuje się za pomocą znaków apostrofu `'` na początku i na końcu:

```
str = 'ciąg_znakowy'
```

Średnik na końcu komendy oznacza wyłączenie echa, czyli pokazania wyniku w linii komend.

Zadanie: Wpisz w linii komend:
`'test';`
Czy powyższy ciąg znakowy został gdziekolwiek zapisany?
Podpowiedź: sprawdź w przestrzeni roboczej.

3.2 Podstawowe działania

Zadanie: Spróbuj wymnożyć macierz b przez wektor a :

$a*b$

Przeczytaj informację o błędzie. Jak mnoży się macierze?

Podstawowe operatory w Matlabie wywołują działania macierzowe. O ile zawsze można wymnożyć skalar i macierz, o tyle macierze muszą mieć odpowiednie wymiary aby dane działanie dało się wykonać. Podstawowe operatory działań zaprezentowano w tabeli 1.

Tablica 1: Podstawowe operatory działań w Matlabie

Operator	Działanie
+	Dodawanie
-	Odejmowanie
*	Mnożenie
/	Dzielenie
^	Potęgowanie
,	Transpozycja

Zadanie: Wykorzystaj odpowiednie operatory aby wyliczyć iloczyny: skalarny i wektorowy zmiennej a przez samą siebie.

Często zamiast działania macierzowego konieczne jest przeprowadzenie takiej samej operacji dla każdego elementu z osobna. W Matlabie w tym celu należy wstawić kropkę przed operatorem:

b^2

$b.^2$

3.3 Definiowanie macierzy

Oprócz możliwości zdefiniowania całej macierzy ręcznie, pewne typy macierzy można zdefiniować za pomocą dedykowanych funkcji. Jak argumenty podaje się w nich kolejno liczbę wierszy i kolumn. Podanie jednej liczby jako argumentu będzie skutkowało powstaniem macierzy kwadratowej. Funkcje te zaprezentowano w tabeli 2.

Tablica 2: Funkcje definiujące macierze w Matlabie

Nazwa	Opis
<code>zeros(w,k)</code>	Macierz zer
<code>ones(w,k)</code>	Macierz jedynek
<code>eye(r)</code>	Macierz jednostkowa
<code>rand(w,k)</code>	Macierz losowych wartości z przedziału $\langle 0, 1 \rangle$

Zadanie: Korzystając z powyższych funkcji stwórz macierz zawierającą same liczby 5 o rozmiarach 4 na 6 (wiersze, kolumny)

3.4 Podstawowe funkcje

Tabela 3 zawiera zestawienie podstawowych funkcji matematycznych w Matlabie.

Tablica 3: Podstawowe funkcje matematyczne w Matlabie

Nazwa	Opis
<code>sin(a)</code> <code>cos(a)</code> <code>tan(a)</code> <code>cot(a)</code>	funkcje trygonometryczne, argument w radianach
<code>sind(a)</code> <code>cosd(a)</code> <code>tand(a)</code> <code>cotd(a)</code>	
<code>sqr(a)</code>	
<code>exp(a)</code>	
<code>log(a)</code>	logarytm naturalny
<code>log10(a)</code>	logarytm o podstawie 10
<code>abs(a)</code>	wartość bezwzględna
<code>pi</code>	zwraca wartość liczby pi, nie przyjmuje argumentów

4 Skrypty

Dłuższe i bardziej skomplikowane obliczenia zapisywane są w skryptach. Są to pliki z rozszerzeniem `*.m` zawierające komendy identyczne z tymi wprowadzanymi w linii komend. Skrypty uruchomić można w dwojaki sposób: za pomocą przycisku `uruchom` (ikona zielonego trójkąta, skrót klawiaturowy F5), na wstążce *Editor* lub poprzez wpisanie nazwy skryptu w linii komend lub innym skrypcie. Nazwy zmiennych i skryptów obowiązują takie same zasady jak dla identyfikatorów w językach C, C++.

Zadanie: Utwórz skrypt w nowym pod-folderze domyślnego folderu Matlab.

W skrypcie zapisz:

```
c = a*b
```

Następnie uruchom skrypt. Jakie pytanie zadaje Matlab/Octave?

Uwaga: W laboratorium zawsze należy wybrać opcję zmiany folderu, nigdy nie należy dodawać folderów do ścieżki!

Matlab uruchamia skrypty i funkcje z plików zapisanych na dysku nie z edytora. Narzędzie *uruchom* zawsze najpierw zapisuje plik na dysku, dopiero potem uruchamia skrypt. Jeżeli wpisana zostanie nazwa nie będąca słowem kluczowym języka i nie będzie obecna w przestrzeni roboczej Matlab rozpoczyna szukanie pliku o takiej nazwie i rozszerzeniu `.m` w folderach zapisanych w zmiennej *path* i w folderze bieżącym. Poszukiwanie kończy się na pierwszym pliku, którego nazwa będzie zgodna. Folder bieżący można zamienić na inny, foldery dodane do ścieżki będą przeszukiwane za każdym razem.

Zadanie: Czy po zamianie folderu skrypt zadziałał? Czy zmienna *c* pojawiła się w przestrzeni roboczej?

Skrypty (zasady nadawania nazw, jak i gdzie Matlab szuka funkcji, folder roboczy, współdzielenie przestrzeni roboczej?)

Skrypty współdzielą przestrzeń roboczą z użytkownikiem. Wszystkie zmienne zdefiniowane w linii komend są dostępne dla skryptów, a zmienne zdefiniowane w skryptach pojawiają się w przestrzeni użytkownika.

Zadanie: Dopisz do skryptu liniijkę:

`z==0`

i uruchom skrypt

W przypadku wystąpienia błędu w skrypcie, informacja o nim zawiera hiperłącze do miejsca w edytorze w którym on wystąpił. Jest to fragment zapisany podkreśloną czcionką.

5 Wykresy

Zadanie: Przepisz poniższy kod do własnego skryptu i uruchom.

```
x=[1 2 3 4];  
y=[1 2 3 4];  
plot(x,y,'-o')  
grid on  
axis tight
```

Do wyświetlania wykresów w Matlabie służy funkcja *plot(x,y)*. Pierwszym argumentem jest wektor zawierający współrzędne odciętych (*x*), a drugim wektor zawierający współrzędne rzędnych (*y*). Kolejne punkty łączone są liniami prostymi

Zadanie: Zmień wektor *y* na następujący:

`y=[1 2 1 2]`

Po obejrzeniu nowego wykresu zmień również wektor *x* na następujący:

`x=[1 2 2 1]`

Funkcja *plot* nie sprawdza czy dany wykres jest funkcją więc możliwe jest rysowanie dowolnych krzywych.

Trzecim argumentem funkcji *plot* jest ciąg formatujący. Może on zawierać specyfikację koloru i typu linii oraz typu markera w punktach danych.

Zadanie: Wpisz w linii komend `help plot` i wciśnij `enter`

W temacie pomocy w linii komend, w pobliżu połowy wysokości można znaleźć wyszczególnienie jakie są możliwe opcje dla ciągu formatującego. Na dole tematu pomocy można znaleźć przykłady wykorzystania sprawdzanej komendy. Poniżej znajduje się hiperłącze do bardziej rozbudowanej pomocy otwieranej w osobnym oknie (Reference page for *plot*). Pomoc taką można otworzyć również wpisując `doc plot`, zamiast `help plot`.

Zadanie: Otwórz temat pomocy w zewnętrznej przeglądarce i zapoznaj się z jej zawartością, zwłaszcza z listą tematów po lewej stronie i działem *See Also* na dole.

Pomoc Matlaba jest dostępna również na stronach internetowych Math-Works, producenta Matlaba.

Kolejne wywołania funkcji `plot` domyślnie nadpisują zawartość umieszczoną w poprzednim oknie rysunków. Można zmienić to za pomocą polecenia:

hold on

Jest ono przełącznikiem zmieniającym tryb działania okna. Po jego użyciu w danym oknie rysunkowym kolejne wywołania funkcji `plot` będą dodawać wykresy w tych samych osiach. Tryb ten można zmienić za pomocą polecenia:

hold off

albo poprzez zamknięcie okna. Nowe okno rysunkowe otworzyć można za pomocą polecenia:

figure

Uwaga: Polecenia `plot`, oraz pokrewne, rysują w ostatnim aktywnym oknie. Jeżeli użytkownik nie zamknie okna z poprzedniego uruchomienia skryptu wykres będzie dodany do poprzedniego okna i zostanie ono przesunięte na wierzch.

Pozostałe polecenia przydatne przy generowaniu wykresów:

- **grid on** – siatka pomocnicza,
- **axis tight** – usunięcie pustych przestrzeni wokół wykresu,
- **axis equal** – równe skale na osiach,
- **title('Tytuł')** – tytuł wykresu,
- **xlabel('x');****ylabel('y')** – etykiety osi,
- **legend('zmienna1','zmienna2',...)** – legenda,
- **xlim([xmin xmax]);ylim([ymin ymax])** – ustawienie zakresów na osiach.

Zadanie: Korzystając w wektora:

`t=1:0.01:1`

Przygotuj w skrypcie wykresy funkcji sinus o częstotliwościach 2 i 3 dla zakresu od 0 do 2π . Kolor linii ustaw na odpowiednio zielony i czerwony, dodaj tytuł wykresu, opisy osi, legendę w położeniu południowy-wschód, siatką pomocniczą i o zakresie osi y między -1.2 a 1.2