

# Wprowadzenie do pakietów MATLAB/GNU Octave

Ireneusz Czajka

wersja poprawiona z 2017

Chociaż dla ścisłości należałoby używać zapisu MATLAB/GNU Octave, w niniejszym opracowaniu używana jest nazwa Matlab, przy czym w zasadzie bez zmian składni można używać tych samych poleceń w pakiecie GNU Octave. Niektóre z podanych przykładów podkreślają równoważność tych pakietów. Ta równoważność dotyczy tylko zakresu podstawowego. Bardziej zaawansowane obliczenia mogą wymagać znacznych modyfikacji skryptów.

## 1. Wprowadzanie danych do przestrzeni roboczej

Dane w Matlab-ie są pamiętane jako zmienne i identyfikowane przez nazwy. Nazwa zmiennej musi zaczynać się od znaku, który nie jest cyfrą ani symbolem działania czy relacji. Zmienne do przestrzeni roboczej można wprowadzać na różne sposoby. Jednym z nich jest wprowadzanie danych bezpośrednio z klawiatury. Poniższe przykłady pomogą zorientować się w sposobach wprowadzania danych do przestrzeni roboczej Matlab-a. Jeżeli wynik działania danej funkcji nie jest przypisany żadnej zmiennej, to Matlab przypisuje go zmiennej o nazwie `ans` i w następnej operacji można korzystać z tej zmiennej.

Na przykład można wpisać następujące wyrażenia w linii poleceń

```
octave:1> [1;3;4]
ans =
```

```
1
3
4
```

```
octave:2> [1 3 4]
ans =
```

```
1 3 4
```

Efektom są dwie macierze: pierwsza ma jedną kolumnę i trzy wiersze, druga jeden wiersz i trzy kolumny. Wniosek: średnik ; powoduje przejście do nowego wiersza. Można zmiennym nadać nazwy na przykład tak:

```
octave:3> a=[1 7 2 1; 2 3 5 4; 3 4 4 6; 4 5 5 1]
a =
```

```
1 7 2 1
2 3 5 4
3 4 4 6
```

```
4 5 5 1
```

```
octave:4> c=[2 7 8 -3; 3 5 4 1;4 4 3 9; 3 3 5 7]
```

```
c =
```

```
2 7 8 -3
3 5 4 1
4 4 3 9
3 3 5 7
```

```
octave:5> d=[1 6 5 1; 1 3 3 6; 2 4 4 8; 3 4 1 1]
```

```
d =
```

```
1 6 5 1
1 3 3 6
2 4 4 8
3 4 1 1
```

Innym sposobem wprowadzania danych jest odczytanie ich z pliku. Zależnie od formatu danych W pliku stosujemy albo odczyt W trybie binarnym, albo odczyt W trybie tekstowym. Poniższy przykład pokazuje jak odczytać dane z pliku tekstowego i binarnego:

```
load plik.dan %odczyt z pliku tekstowego
load plik.mat %odczyt z pliku binarnego
```

Ciekawym i użytecznym sposobem tworzenia macierzy w Matlab-ie i odwoływania się do ich elementów jest tzw. notacja dwukropkowa, dzięki której można w zwięzły sposób generować wektory elementów o stałym kroku:

—  $j:k$  taki zapis generuje wektor  $[j, j+1, j+2, \dots, k]$

—  $j:h:k$  taki zapis generuje wektor podobny do powyższego, z tym, że kolejne wartości elementów wektora są zwiększane o  $h$ , a nie o 1 jak wyżej; czyli uzyskuje się wektor  $[j, j+h, j+2h, j+3h, \dots, k]$

Korzystając z notacji dwukropkowej można też dokonywać wyboru kolumn lub Wierszy z macierzy, aby dokonywać na nich operacji.

$A(:,i)$  wypisanie  $i$ -tej kolumny macierzy  $A$

$A(:,j:k)$  wypisanie kolumn od  $j$  do  $k$  macierzy  $A$

$A(1,:)$  wypisanie  $i$ -tego wiersza macierzy  $A$ , np.:

```
>> y=0:.1:1
```

```
y =
```

```
Columns 1 through 7:
```

```
0.00000 0.10000 0.20000 0.30000 0.40000 0.50000 0.60000
```

```
Columns 8 through 11:
```

```
0.70000 0.80000 0.90000 1.00000
```

```
>> A=magic(5)
```

```
A =
```

```

17 24 1 8 15
23 5 7 14 16
4 6 13 20 22
10 12 19 21 3
11 18 25 2 9

```

```
>> y=A(:,3)
```

```
y =
```

```

1
7
13
19
25

```

Więcej przykładów można znaleźć na przykład w pozycji [1] na stronach 62 i 63.

## 2. Operacje macierzowe w Matlabie

Ponieważ domyślnym przedmiotem operacji wykonywanych w Matlab-ie są macierze, więc każda zmienna czy stała użyta przez użytkownika w trakcie sesji jest traktowana jak macierz. W Matlab-ie zaimplementowano dwa rodzaje operacji na macierzach

- arytmetyczne operacje macierzowe, określone regułami algebry liniowej;
- arytmetyczne operacje tablicowe, wykonywane na elementach macierzy.

Tabela 1 zawiera zestawienie podstawowych operatorów macierzowych i tablicowych zaimplementowanych w Matlab-ie Oprócz operatorów arytmetycznych Matlab zawiera operatory logiczne i operatory relacji. Służą one do dokonywania porównań. Tabela 2 zawiera spis tych operatorów dostępnych w opisywanym środowisku.

Tabela 1. Operatory arytmetyczne

Operator macierzowy	Opis	Operator tablicowy	Przykład macierzowy	Przykład tablicowy	Uwagi
+	Dodawanie	+	A+B	A+B	plus
-	Odejmowanie	-	A-B	A-B	minus
*	Mnożenie	*	A*B	A.*B	mtimes, times
^	Potęgowanie	.^	A^B	A.^B	mpower, power
\	Dzielenie lewostronne	.\	A\B	A.\B	mldivide, ldivide
/	Dzielenie prawostronne	./	A/B	A./B	mrdivide, rdivide
'	Sprzężenie hermitowskie		A'		dla macierzy rzeczywistych jest transpozycją
.'	Transpozycja		A.'		ctranspose transpose

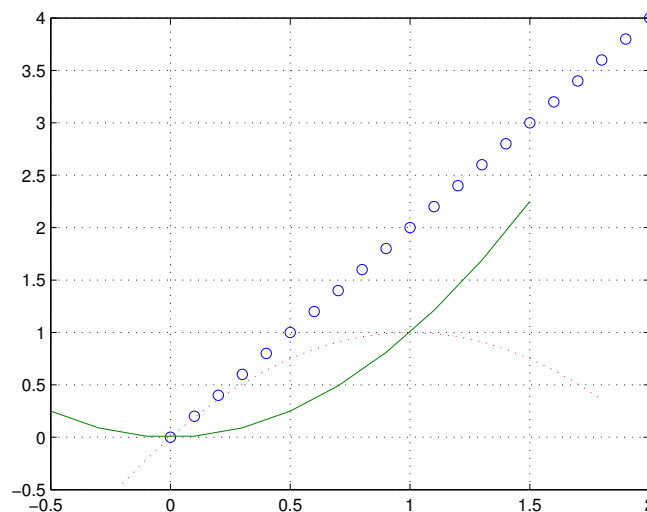
Tabela 2. Operatory logiczne i operatory relacji

Symbol	Relacja	Symbol	Operacja logiczna
<	mniejsze od	&	AND – koniunkcja
<=	mniejsze lub równe		OR – alternatywa
>	większe od	~	NOT – negacja
>=	mniejsze lub równe	xor	wyłącznie OR
==	równe		
~=	różne od		

### 3. Podstawy użycia grafiki w Matlab-ie

Wszystkie wyniki obliczeń można w łatwy sposób obrazować graficznie. Niektóre służące do tego celu funkcje zostały zestawione w tabeli 4. Jako argumenty większości z tych funkcji można podawać rodzaj i kolor linii, którą ma być rysowany dany wykres. Można w jednym poleceniu podać kilka zestawów danych do rysowania, aby zostały zobrazowane na jednym wykresie np.:

```
x1=0:0.1:2;
y1=2*x1;
x2=-0.5:0.2:1.5;
y2=x2.^2;
x3=x1-0.2;
y3=2*x3-x3.^2;
plot(x1, y1,'o', x2, y2, x3, y3, ':'')
grid
```



Rysunek 1. Przykładowy rysunek

Jak pokazano na rysunku 1, powyższe polecenia spowodują narysowanie trzech zestawów danych na jednym wykresie w następujący sposób

- $y_1=f(x_1)$  zaznaczy punkty okręgami,
  - $y_2=f(x_2)$  łączy punkty stosując interpolację liniową,
  - $y_3=f(x_3)$  zostanie przedstawiony linią punktową,
- oprócz tego zostanie narysowana siatka współrzędnych.

Kolory zostaną dobrane automatycznie w taki sposób aby odróżnić wykresy od siebie. Inne rodzaje linii zostały przedstawione w tabeli 3.

Tabela 3. Rodzaje linii w Matlab-ie

Symbol	Rodzaj linii	Symbol	Rodzaj linii	Symbol	Rodzaj linii
.	punkt	+	plus	:	kropkowa
o	okrąg	*	gwiazdka	-.	punktowa
x	znak x	-	ciągła	--	kreskowa

Tabela 4. Wybrane funkcje graficzne

Polecenie	Opis	Przykład
<code>plot</code>	skala liniowa na osiach	<code>x=1:10;y=sin(x);plot(x,y)</code>
<code>loglog</code>	skala logarytmiczna na osiach	<code>x=1:10;y=sin(x);loglog(x,y)</code>
<code>bar</code>	wykres słupkowy	<code>x=[1 2 5];y=[1 3 2];bar(x,y)</code>
<code>hist</code>	histogram	<code>y=rand(100,1);hist(y,20)</code>
<code>rose</code>	histogram kołowy	<code>rose(randn(1,1000))</code>
<code>semilogx</code>	skala logarytmiczna na osi x	<code>x=1:10;y=sin(x);semilogx(x,y)</code>
<code>semilogy</code>	skala logarytmiczna na osi y	<code>x=1:10;y=sin(x);semilogy(x,y)</code>
<code>boxplot</code>	wykres pudełkowy	<code>x=randn(100,5);boxpot(x)</code>
<code>fplot</code>	wykres funkcji ciągłej	<code>fplot(7sin(x)', [0 4*pi])</code>
Opisy i komentarze wykresów		
<code>grid</code>	nałożenie siatki współrzędnych	<code>fplot(7sin(x)', [0 B]); grid</code>
<code>gtext</code>	umieszczenie napisu w miejscu wskazanym myszą	<code>gtext('Ten tekst wstawiam')</code>
<code>text</code>	umieszczenie tekstu w zadanym miejscu wykresu	<code>x=1:10;comet(x,x.^2); text(1,1,'To wstawiamy')</code>
<code>title</code>	tytuł wykresu	<code>fplot('sin(x)', [0 3]);title('Tytuł')</code>
<code>xlabel</code>	opis osi x	<code>fplot('sin(x)', [0 3]);xlabel('Pozioma')</code>
<code>ylabel</code>	opis osi y	<code>fplot('sin(x)', [0 3]);ylabel('Pionowa')</code>

Warto pamiętać o tym, że Matlab pozwala na eksportowanie rysunków w wielu formatach. Do zapisu bieżącego aktywnego rysunku służy polecenie

```
print('-depsc2', 'rysunek1.eps')
```

rysunek zostanie zapisany w formacie eps. Oprócz tego formatu można wykorzystać wiele innych. Możliwości polecenia `print` zostały szerzej opisane w pomocy programu.

#### 4. Przegląd możliwości Matlab-a za pomocą programów demonstracyjnych

Niniejszy podrozdział zawiera treści dotyczące nieco starszych wersji Matlab-a.

- `expomap` – główny plik demonstracyjny Matlab—a;

Tabela 5. Podstawowe polecenia do pracy z plikami i katalogami

Polecenie	opis
<code>dir, ls</code>	wyświetla zawartość katalogu
<code>type</code>	wypisuje zawartość pliku na ekran
<code>cd</code>	pozwala na zmianę bieżącego katalogu
<code>pwd</code>	wyświetla ścieżkę bieżącego katalogu
<code>help</code>	wyświetla komentarze z początku m-pliku jako pomoc do pliku

- `intro` –przeгляд właściwości Matlab—a, jeżeli chodzi o operacje macierzowe, wprowadzanie zmiennych do przestrzeni roboczej;
- `matmanip` – operacje macierzowe;
- `graf2d` – wykresy dwuwymiarowe w Matlab-ie
- `funfuncs` – podstawowe operacje algebraiczne na funkcjach
- `grafcplx` – wykresy funkcji zespolonych
- `xplang` – przykłady m-plików funkcyjnych

## 5. Polecenia dotyczące obsługi plików i katalogów

Czasami zachodzi potrzeba wykonania operacji na plikach lub katalogach bez opuszczania sesji Matlab—a. Podstawowym przykładem jest ustalenie nowego katalogu roboczego. Aby tego dokonać korzystamy z polecenia `cd ścieżka_dostępu`

Tabela 5 zawiera wybrane polecenia dotyczące pracy z plikami i katalogami. Można też wykorzystywać polecenia systemu operacyjnego za pomocą operatora `!` lub jako parametry polecenia `dos` czy `unix` np.:

```
!date,
dos('p2mope.exe')
lub unix('ls -lah').
```

## Literatura

- [1] B. Mrozek, Z. Mrozek: MATLAB uniwersalne środowisko do obliczeń naukowo-technicznych. PLJ Warszawa 1996
- [2] A. Zalewski, R. Cegiela: MATLAB – obliczenia numeryczne i ich zastosowania. Wydawnictwo Nakom Poznań 1996
- [3] P. Drozdowski: Wprowadzenie do MATLAB-a. Politechnika Krakowska, Kraków 1995