helion.pl
KSIĘGARNIA INTERNETOWA · WYDAWNICTWO

IDŹ DO
PRZYKŁADOWY ROZDZIAŁ
SPIS TREŚCI

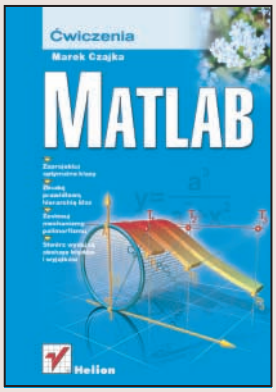
KATALOG KSIĄŻEK
KATALOG ONLINE
ZAMÓW DRUKOWANY KATALOG

TWÓJ KOSZYK
DODAJ DO KOSZYKA

CENNIK I INFORMACJE
ZAMÓW INFORMACJE O NOWOŚCIACH
ZAMÓW CENNIK

CZYTELNIA
FRAGMENTY KSIĄŻEK ONLINE

MATLAB. Ćwiczenia



Autor: Marek Czajka
ISBN: 83-7361-711-6
Format: B5, stron: 96
Przykłady na ftp: 2 kB

Matlab to środowisko służące do przeprowadzania obliczeń numerycznych i rozwiązywania złożonych zagadnień matematycznych. Wydajne i szybkie algorytmy oraz doskonałe mechanizmy analityczne czynią z niego doskonałe narzędzia zarówno dla matematyka, jak i dla ekonomisty lub genetyka. Język programowania Matlaba umożliwia tworzenie własnych aplikacji, a ogromna ilość bibliotek zewnętrznych ułatwia zastosowanie tych aplikacji do różnych celów. Matlab to dziś standard w laboratoriach badawczych. Coraz częściej wykorzystuje się go również na uczelniach – na niektórych jest nawet elementem programu zajęć. Oczywiście nic nie stoi na przeszkodzie, aby jego możliwości wykorzystywać również podczas nauki matematyki w szkole.

Książka „Matlab. Ćwiczenia" to zbiór praktycznych ćwiczeń, dzięki którym poznasz możliwości Matlaba i zasady korzystania z niego. W książce opisane jest tworzenie programów, wykorzystywanie Matlaba do obliczeń matematycznych – zarówno podstawowych jak i tych najbardziej skomplikowanych oraz tworzenie graficznej reprezentacji wyników obliczeń.

- Środowisko Matlaba
- Zasady programowania w Matlabie
- Funkcje matematyczne
- Operacje na macierzach
- Wyznaczanie pochodnych i całek
- Sterowanie przebiegiem programu
- Tworzenie różnych typów wykresów

Przekonaj się, jak bardzo możesz przyspieszyć swoją pracę korzystając z odpowiedniego narzędzia obliczeniowego.

Wydawnictwo Helion
ul. Chopina 6
44-100 Gliwice
tel. (32)230-98-63
e-mail: helion@helion.pl

Spis treści

Wstęp **5**

Rozdział 1. Praca w MATLAB-ie **7**

Uruchamianie MATLAB-a 7
Program 8
Edytor M-plików 10
Zamykanie MATLAB-a 12

Rozdział 2. Programowanie w MATLAB-ie **13**

Zmienne 13
Pomocne zmienne MATLAB-a 17
Operatory 18
Funkcje 19
Transpozycja 20
Informacje o zmiennych 21
Wartość bezwzględna 22
Zaokrąglanie 23
Funkcje trygonometryczne 24
Funkcje logarytmiczne 25
Pierwiastek 27
Losowanie wartości 28
Formaty liczb 31
Sumowanie elementów 33
Macierz dolno- i górnoprzekątna 34
Kwadrat magiczny 35
Macierz Hilberta 36
Macierz Pascala 37
Macierz Vandermonde'a 37
Macierze elementarne 38
Wyznacznik 38
Informacje o poleceniach 39
Wyświetlanie tekstu 42
Funkcje Inline 42
Pochodna funkcji 43
Wyliczanie całek 44
Odmierzanie czasu 46

Instrukcja warunkowa ... 46
Instrukcja wyboru ... 48
Pętle ... 50
Pętla for ... 50
Pętla while ... 51
Obsługa błędów ... 52
Komentarze ... 53
Pisanie funkcji i skryptów ... 53
Przykłady pisania funkcji i skryptów ... 54
Wczytanie zmiennych ... 54
Największy wspólny dzielnik ... 54
Najmniejsza wspólna wielokrotność ... 56
Zmiana systemu liczbowego ... 59
Gra ... 61

Rozdział 3. Podstawy tworzenia wykresów ... 65

Okno wykresów ... 66
Pierwszy wykres ... 67
Wzbogacanie wykresu ... 71
Wykres kołowy ... 77
Histogram ... 79
Wykres słupkowy ... 81
Wykres schodkowy ... 84
Wiele wykresów w jednym oknie ... 85
Wykresy demonstracyjne ... 87

Rozdział 4. MATLAB w internecie ... 89

Rozdział 2.

Programowanie w MATLAB-ie

Na początku tego rozdziału zajmiemy się podstawowymi poleceniami MATLAB-a. Zobaczymy, w jaki sposób tworzy się nowe zmienne oraz jak program je traktuje. W międzyczasie przyjrzymy się wielu ciekawym funkcjom, których w tym środowisku nie brakuje.

W dalszej części rozdziału przejdziemy do programowania. Zajmiemy się pisaniem funkcji i skryptów o różnym stopniu skomplikowania.

Zmienne

Zmienne to nic innego, jak tylko pewna informacja zapisana w pamięci komputera. MATLAB daje możliwość operowania zmiennymi wielowymiarowymi, o czym przekonasz się niebawem.

Na początku przejdź do okna *Command Window* i wpisz tam pewną wartość liczbową (np. 13), a następnie zatwierdź, używając klawisza Enter. Automatycznie w oknie poleceń pojawi się informacja:

```
ans =

    13
```

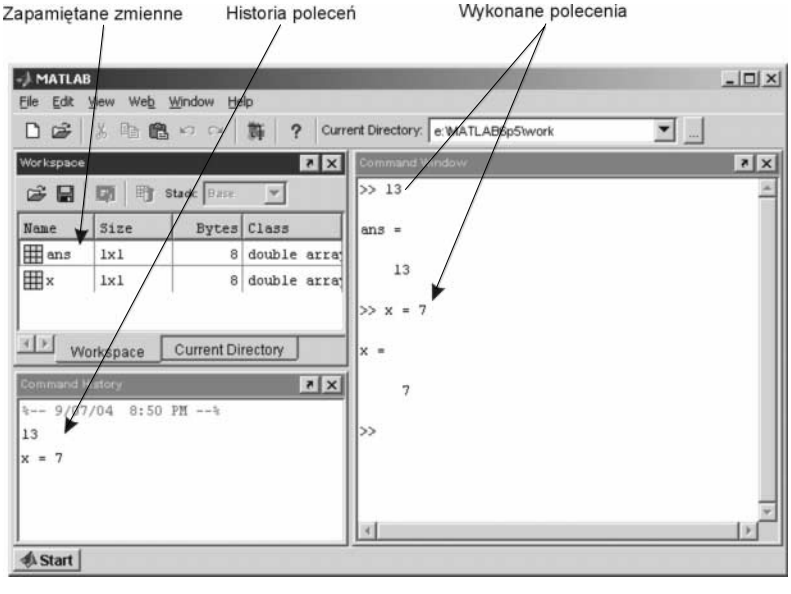
która mówi, że zmiennej `ans` przyporządkowana jest wartość `13`. MATLAB sam stworzył tę zmienną. Przypisanie pewnej wartości do nowej bądź istniejącej zmiennej zaprezentowane jest poniżej.

Za znakiem zachęty (>>) wpisz `x = 7`, a następnie zatwierdź klawiszem Enter. Poniżej zostanie wyświetlona informacja:

```
x =

     7
```

Od tej chwili zmiennej `x` przypisana jest wartość `7`. Na rysunku 2.1 przedstawiono wykonane polecenia.



Rysunek 2.1. *Wykonywanie poleceń*

Podczas przypisywania wartości zmiennym można wykonywać obliczenia lub całe wyrażenia.

```
>> y = 12 + 7

y =

    19

>> z = 23 - 10

z =

    13

>> x = y - z

x =

     6

>> x = x * 3

x =

    18
```



Zauważ, że wyrażenie może zawierać wcześniej zadeklarowane zmienne.

Teraz zajmiemy się deklarowaniem zmiennych wielowymiarowych. Nie jest to trudne, wystarczy pamiętać o tym, żeby tworzone macierze były prostokątne.

Przykładowo, stworzymy zmienną o rozmiarze 4 na 3. W tym celu należy użyć polecenia:

```
X = [1 2 3 4; 5 6 7 8; 9 10 11 12]
```

W ten sposób otrzymaliśmy zmienną o następującej budowie:

```
     1     2     3     4
     5     6     7     8
     9    10    11    12
```

Jak widać, kolejne wartości w wierszach oddziela się spacjami, natomiast przechodzenie do następnej linii następuje po wpisaniu znaku średnika.



Zauważ, że deklarowanie zmiennych różnych typów niczym się nie różni. Dotyczy to także deklarowania zmiennych przechowujących ciągi znaków.

Deklarowanie zmiennej, której wartością jest ciąg znaków, wygląda następująco:

```
>> s = 'Matlab jest ok!'

s =

Matlab jest ok!
```

MATLAB bez problemu rozpoznaje również liczby zespolone. Wie także, jakiego typu jest nowa zmienna!

Za pomocą znaku dwukropka można tworzyć łańcuchy liczb, np. `1,2,3,4` lub `4, 2, 0`. Konstrukcja wykorzystująca dwukropek wygląda następująco:

```
j:i:k
```

Generuje ona następujący ciąg znaków:

```
[j, j+i,j+2i,…,k]
```

W przypadku gdy `i=1`, można zapisać:

```
j:k
```

Jeżeli wielkość `i` nie jest określona, MATLAB domyślnie przyjmuje, że `i=1`.

Oto przykłady tworzenia ciągów:

```
>> x=1:3

x =

     1     2     3

>> 4:-2:0
```

```
ans =

     4     2     0

>> 0:pi:10

ans =

         0    3.1416    6.2832    9.4248
```

Aby móc odczytywać dane z tablicy, można posłużyć się następującymi metodami (użyjemy tu konstrukcji `A = [1 2 3 4; 5 6 7 8; 9 10 11 12]`):

- *wskazanie jednej wartości* — `A(x,y)`, gdzie `x` oznacza numer wiersza, a `y` numer kolumny; np.:

```
>> A(2,3)

ans =

     7
```

- *wskazanie jednej kolumny* — `A(:,y)`, gdzie `y` oznacza numer kolumny; np.:

```
>> A(:,3)

ans =

     3
     7
    11
```

- *wskazanie jednego wiersza* — `A(x,:)`, gdzie `x` oznacza numer wiersza; np.:

```
>> A(2,:)

ans =

     5     6     7     8
```

- *wskazanie wszystkich danych w formie wektora* — `A(:)`; np.:

```
>> A(:)

ans =

     1
     5
     9
     2
     6
    10
     3
     7
    11
     4
     8
    12
```

❖ *wskazanie macierzy wewnętrznej* — `A(x1:krok_x:x2,y1:krok_y:y2)`, gdzie `x1`, `x2` to numery wierszy, natomiast `y1`, `y2` to numery kolumn; np.:

```
>> A(2:3,3:4)

ans =

     7     8
    11    12
```



Pamiętaj, że za pomocą symboli `()` określasz indeksy tablic, natomiast za pomocą symboli `[]` tworzysz tablice.

Tworzenie zmiennych

Ćwiczenie 2.1.

W tym ćwiczeniu zapoznasz się z tworzeniem zmiennych i odczytywaniem ich wartości. Aby nauczyć się obsługi zmiennych:

1. W oknie *Command Window* wpisz `15` i zatwierdź klawiszem Enter. Zostanie stworzona zmienna o nazwie `ans`, której wartością jest właśnie `15`.
2. Stwórz zmienną `x` i przypisz jej wartość `3`.
3. Do zmiennej `a` przypisz wartość działania `x*3`. W tym celu wykonaj polecenie `a=x*3`.
4. Stwórz zmienną o rozmiarze 3 na 3. Możesz posłużyć się następującym poleceniem: `A=[1 2 3; 2 3 4; 1 3 4]`.
5. Przypisz zmiennej `x` wartość przechowywaną w środku macierzy `A`. W tym celu użyj polecenia `x=A(2,2)`.

Pomocne zmienne MATLAB-a

MATLAB posiada wiele „wbudowanych” zmiennych, które mogą być wykorzystywane przez użytkownika w dowolnym momencie. Dostarczają one wartości liczbowych często stosowanych przez programistów korzystających z tego środowiska. Oto niektóre z tych zmiennych:

❖ `pi` — wartość PI;

❖ `date`, `clock` — aktualna data i czas;

❖ `NaN` — wartość nieokreślona;

❖ `Inf` — nieskończoność.

Przykładowo, aby odczytać aktualną datę i przypisać ją do zmiennej, wystarczy użyć polecenia `x=date`. Format otrzymanych danych po wydaniu polecenia `clock` jest następujący:

```
>> clock

ans =

  1.0e+003 *

    2.0040    0.0090    0.0090    0.0230    0.0310    0.0215
```

Odczytywanie wartości zmiennych pomocniczych

Ćwiczenie 2.2.

Aby zapoznać się ze zmiennymi pomocniczymi:

1. Do zmiennej `x` przypisz wartość `pi`.
2. Do zmiennej `y` przypisz wartość `date`.
3. Do zmiennej `z` przypisz wartość `clock`.
4. Wyświetl wartości tych zmiennych za pomocą polecenia: `x,y,z`.

Operatory

MATLAB pozwala stosować szereg operatorów logicznych, operatorów relacji oraz operatorów arytmetycznych. Poniżej są one kolejno przedstawione.

Operatory logiczne — są stosowane przy budowaniu wyrażeń logicznych. Symbol *&* oznacza „i", symbol | oznacza „lub", ~ oznacza „nie" (negację), natomiast *xor* oznacza „alternatywę wykluczającą".

Operatory relacji — pozwalają porównywać ze sobą dwie wartości: <, <=, >, >=, ==, ~= (różny).

Operatory arytmetyczne — pozwalają wykonywać operacje pomiędzy wartościami. W MATLAB-ie istnieje podział operatorów na macierzowe i tablicowe. Operatory arytmetyczne to: +, –, *, .*, ^, .^, /, ./, \, .\.

Symbole poprzedzone kropką odnoszą się do operacji tablicowych.

Operacje na macierzach

Ćwiczenie 2.3.

Aby zapoznać się z niektórymi z operatorów, proponuję wykonać następujące ćwiczenie:

1. Wykonaj polecenie: `a=[1 2; 3 4]`, a następnie polecenie `b=[2 3; 4 5]`. Dzięki temu MATLAB zapamięta takie wartości:

```
a =

     1     2
     3     4


b =

     2     3
     4     5
```

2. Aby dodać te dwie macierze i wynik zachować w następnej, wykonaj polecenie: `c = a + b`. Wartość nowej zmiennej jest następująca:

```
c =

     3     5
     7     9
```

3. Aby pomnożyć macierze przez siebie, wpisz: `c = a * b`, a następnie użyj klawisza Enter.

4. Aby podnieść macierz do trzeciej potęgi, należy wykonać polecenie: `c = a ^3`

```
c =

    37    54
    81   118
```



Jeśli zakończysz polecenie znakiem średnika, to efekt jego działania nie zostanie wyświetlony w oknie poleceń.