

Ćwiczenie Mat

Wprowadzenie do Matlaba

Część 1. Zmienne i wyrażenia, operacje na tablicach

Polecenia w Matlabie możemy wpisywać bezpośrednio w oknie poleceń lub przygotowywać zewnętrzne pliki tekstowe z całą sekwencją poleceń (tzw. skrypty lub m-pliki). Pliki te można tworzyć we wbudowanym w Matlabie edytorze.

Przykłady instrukcji Matlaba

%abc... – komentarz w Matlabie.

a = 5; - nadanie zmiennej a wartości 5. Jeśli pominiemy znak ; wartość tej zmiennej wyświetli się.

log(a) – wywołanie funkcji bibliotecznej z argumentem a.

b = sin(pi/2) – zapamiętanie w zmiennej b wyniku działania funkcji sinus z argumentem pi/2.

c = [1 2 3] – utworzenie wektora o trzech elementach.

d = [1 2; 3 4; 5 6] – utworzenie macierzy d o wymiarach 3x2. ; oznacza tu oddzielenie wierszy. Elementy tego samego wiersza oddziela się spacjami lub przecinkami.

d(1,2) – odwołanie do elementu w 1-szym rzędzie i 2-jej kolumnie. Np. d(1,2) = 5, a = d(1,2).

d(end,end) – odwołanie do elementu w ostatnim rzędzie i ostatniej kolumnie.

d(:,1) – odwołanie do wszystkich elementów 1-wszej kolumny macierzy d. Np. x = d(:,1) daje x = [1; 2; 5], d(:,1) = [0; 0; 0] daje d = [0 2; 0 4; 0 6].

d(2,:) – odwołanie do wszystkich elementów 2-go wiersza macierzy d. Np. x = d(2,:) daje x = [3 4], d(2,:) = [0 0] daje d = [1 2; 0 0; 5 6], d(1,:) = d(2,:) skopiowanie 2-go wiersza macierzy d do 1-go.

d(1:2,:) – odwołanie do wszystkich elementów 1-go i 2-go wiersza macierzy d.

d([1 3],:) – odwołanie do wszystkich elementów 1-go i 3-go wiersza macierzy d.

d(2:3,1:2) – odwołanie do elementów z 2-go i 3-go wiersza oraz z 1-wszej i 2-jej kolumny.

e = 1:10 – utworzenie wektora o elementach [1 2 ... 10].

f = -5:2:5 – utworzenie wektora f o pierwszym elemencie -5, drugim -5+2, trzecim -5+2*2, itd. Ostatni element <= 5.

f(1) – odwołanie do 1-go elementu wektora f (np. f(1) = 2, a = f(1)).

size(d) – funkcja zwracająca rozmiary macierzy (wektora) – liczbę wierszy i liczbę kolumn; size(d,1) zwraca liczbę wierszy; size(d,2) zwraca liczbę kolumn.

length(c) – funkcja zwracająca długość wektora c.

s = 'Matlab' – utworzenie łańcucha znaków.

`[]` – macierz pusta, służy do usuwania wierszy lub kolumn macierzy, np. `d(1,:) = []` spowoduje usunięcie 1-go wiersza macierzy `d`.

`A = [B; C]` – utworzenie macierzy `A` z macierzy `B` i `C` w ten sposób, że do każdej kolumny macierzy `B` dopisywane są odpowiednie kolumny macierzy `C`. Macierze `B` i `C` muszą mieć tę samą liczbę kolumn.

`A = [B C]` – utworzenie macierzy `A` z macierzy `B` i `C` w ten sposób, że do każdego wiersza macierzy `B` dopisywane są odpowiednie wiersze macierzy `C`. Macierze `B` i `C` muszą mieć tę samą liczbę wierszy.

`A = rand(3, 4)` – utworzenie macierzy o rozmiarach 3×4 i zapełnienie jej liczbami losowymi z rozkładu jednostajnego $0 - 1$.

`A = rand(4)` – utworzenie macierzy o rozmiarach 4×4 i zapełnienie jej liczbami losowymi z rozkładu jednostajnego $0 - 1$.

`A = zeros(3, 4)` – utworzenie macierzy o rozmiarach 3×4 i zapełnienie jej zerami.

`A = ones(3, 4)` – utworzenie macierzy o rozmiarach 3×4 i zapełnienie jej jedynekami.

`A+B`, `A.*B` – dodanie lub pomnożenie odpowiadających sobie elementów macierzy `A` i `B` (macierze muszą mieć jednakowe rozmiary).

`clear` – usunięcie wszystkich zmiennych ze środowiska.

`save plik.txt a -ascii` – zapisanie zmiennej `a` do pliku tekstowego `plik.txt`.

`load plik.txt` – załadowanie zmiennej z pliku `plik.txt`.

`help nazwa_m_pliku` – pomoc na temat funkcji bibliotecznej lub m-pliku, np. `help sin`.

`Ctrl+C` – przerwanie działania uruchomionego programu.

Zadania do wykonania

- 1.1. Przećwicz w środowisku Matlaba powyższe polecenia.
- 1.2. Utwórz w dowolny sposób macierze `A` i `B` o wymiarach 4×4 (każdy student tworzy inne macierze).
- 1.3. Wykonaj na tych macierzach podstawowe operacje arytmetyczne: `A+B`, `A-B`, `A*B`, `A.*B`, `A/B`, `A./B`, `A\B`, `A.\B`, `A^2`, `A.^B`, `A'`, `A.'` (opis operacji znajdziesz w helpie Matlaba lub Internecie).
- 1.4. Pobierz z macierzy `A` pierwszy wiersz (nazwij ten wiersz `X`) i z macierzy `B` 2-gą i 3-cią kolumnę (nazwij tę podmacierz `Y`). Podaj polecenia Matlaba, które wykonają następujące działania oraz wyniki tych działań:
 - a. $3 * X * 5 * Y$
 - b. $(4 * X^T * X)^2 * 2 * Y$
 - c. odejmij od każdego elementu macierzy `Y` liczbę π i pomnóż każdy element macierzy wynikowej przez ostatni element wektora `X`. Transponuj tak otrzymaną macierz otrzymując macierz `Z`. Następnie utwórz nową macierz `Q`, łącząc macierz `Z` z transponowaną macierzą `Y`, w ten sposób, aby macierz `Z` wstawiona była w 1 i 2 wierszu macierzy `Q`, a transponowana macierz `Y` wstawiona była w 3 i 4 wierszu macierzy `Q`

Część 2. Instrukcje i funkcje

Przykłady instrukcji Matlab

sum(A) – suma elementów wektora lub, jeśli A jest macierzą, suma poszczególnych kolumn macierzy.

[wart,ind] = max(A) – dla wektorów maksymalna wartość elementu wektora (wartość ta zapamiętana będzie w zmiennej wart, a indeks elementu o największej wartości w zmiennej ind). Dla macierzy – zwraca maksymalne wartości elementów w każdej kolumnie. Analogicznie działa funkcja min().

```
for i=1:5
    %polecenia, które wykonają się w pętli
end
```

- pętla for wykonująca się pięciokrotnie (i=1:5), inny przykład: for j = 2.1:-0.1:-1.5 ... end – pętla wykonująca się dla kolejnych wartości licznika j = 2.1, 2.0, 1.9, ..., -1.5.

```
while a > 0
    % polecenia, które wykonają się w pętli
end
```

- pętla while wykonująca się pod warunkiem a>0

```
if b == 0
    %polecenia, które wykonają się, jeśli b jest równe 0 (== oznacza równość logiczną)
end
```

```
if b ~= 0
    %polecenia, które wykonają się, jeśli b nie równa się 0
else
    %polecenia, które wykonają się, jeśli b równa się 0
end
```

fprintf('Wartość zmiennej: %5.2f \n', x) – funkcja wyjścia wyświetlająca komunikaty na monitorze. Ciąg wyświetlanych znaków podajemy w apostrofach. %5.2f – sekwencja formatująca, zamiast której podstawiona będzie wartość zmiennej x. 5.2 oznacza, że liczba wyświetlona będzie na 5-ciu pozycjach (jeśli jest krótsza zostanie dopełniona spacjami, jeśli jest dłuższa od 5 wartość ta jest ignorowana) z dokładnością do 2-ch miejsc po przecinku. f oznacza liczbę zmiennopozycyjną. \n to znak przejścia do następnej linii.

NaN – not a number; symbol oznaczający w Matlabie wartość nienumeryczną, otrzymaną np. w wyniku działania 0.0/0.0.

Inf - symbol oznaczający w Matlabie nieskończoność.

plot(x,y) – tworzenie wykresu; x – wektor współrzędnych x punktów, y – wektor współrzędnych y punktów. Np. x=-pi:0.1:pi; plot(x,sin(x)). Jeśli x i y to macierze tworzonych jest tyle wykresów ile jest kolumn.

figure(x) – tworzenie nowego okna do wykresu; x oznacza numer okna.

hold on – powoduje, że kolejne wykresy rysowane są na poprzednio narysowanych. Hold off powoduje skasowanie wcześniejszych wykresów.

W Matlabie możemy tworzyć własne funkcje zapisując je w oddzielnych m-plikach. Np. poniżej podano definicję funkcji obliczającej pierwiastek równania liniowego. Definicja zaczyna się słowem function, dalej jest nazwa zmiennej zwracanej przez funkcję (jeśli zmiennych jest więcej wymieniamy je w nawiasach kwadratowych, np. [x, y, z]) i nazwa funkcji (uwaga plik należy nazwać tak samo jak nazwa funkcji, w tym wypadku rown1.m). W nawiasach wymieniamy argumenty funkcji (tu – współczynniki równania prostej). W kolejnych liniach piszemy instrukcje (tzw. ciało funkcji), kończąc definicję funkcji słowem end.

```
function x = rown1(a, b)
if a == 0
    if b ~= 0
        x = []; %jeśli a=0 i b<>0 nie zwracaj niczego (brak rozwiązania)
    else
        x = NaN; %jeśli a=0 i b=0 (nieskończenie wiele rozwiązań)
    end;
else
    x = -b / a; %jeśli a<>0 można wyznaczyć pierwiastek
end;
```

Przykładowe wywołania funkcji: `p = rown1(3,9)`, `v = rown1(a1, a2)`;

Jeśli funkcja zwraca więcej niż jedną wartość wywołanie ma postać: `[a, b, c] = nazwa_funkcji(d, e, f, g)`;

Po wykonaniu instrukcji (wpisanych w oknie poleceń lub zawartych w wykonywanym skrypcie) utworzone w ich wyniku zmienne dostępne są cały czas w środowisku Matlaba (do zamknięcia programu). Możemy podejrzeć ich wartości wpisując nazwy zmiennych.

Skrypty uruchamiamy wpisując w oknie poleceń nazwę skryptu bez rozszerzenia.

Uwaga: zmienne występujące w ciele funkcji traktowane są jak zmienne lokalne; po wykonaniu funkcji znikają.

Zadania do wykonania

- 2.1. Przećwicz w środowisku Matlaba powyższe polecenia.
- 2.2. Zapoznaj się z operatorami relacji (`>`, `==`, `~=`, ...) i z operatorami logicznymi (`&`, `|`, `&&`, `||`). Podaj przykłady użycia wszystkich tych operatorów.
- 2.3. Podaj sensowne przykłady użycia pętli `for` i `while` oraz instrukcji warunkowej.
- 2.4. Napisz funkcję, która oblicza pierwiastki równania kwadratowego i rysuje parabolę w zakresie od `x1` do `x2` (`x1` i `x2` to pierwiastki). Podaj przykład jej użycia (wywołanie).

Zawartość sprawozdania

Sprawozdania powinny być sporządzone według wzoru zamieszczonego na stronie i zawierać:

- A) Cel ćwiczenia.
- B) Treść zadania.
- C) Krótki opis środowiska i języka Matlab.
- D) Metodyka rozwiązania zadania.
- E) Raport z p. 1.2-1.4 (polecenie Matlab, wynik działania polecenia i komentarz, np.

%dodawanie dwóch macierzy o identycznych rozmiarach:

C=A+B

C =

1.2365	1.2881	1.6362	1.6126
1.8215	0.1333	1.7226	0.6566
0.9192	1.1276	0.9007	1.5063
1.8729	1.4809	1.3628	0.1737

Raport z p. 2.2-2.4 - polecenie Matlab, wynik działania polecenia i komentarz, np.

```
%Przykład użycia operatora logicznego lub (||). Jeśli podana liczba jest spoza zakresu, zgłoś błąd
x = input('Podaj liczbę z zakresu od 1 do 10:');
if(x<1)|| (x>10)
    disp('error');
end
```

- F) Wnioski (możliwości Matlab, specyficzne cechy Matlab, instrukcje złożone w Matlabie, funkcje w Matlabie, czego nauczyłeś się realizując to ćwiczenie itp.)

Przykładowe zagadnienia i pytania zaliczeniowe

Polecenia i instrukcje Matlab z tego ćwiczenia

Do przygotowania na następne zajęcia

1. Zapoznać się z instrukcją do kolejnego ćwiczenia.
2. Zapoznać się z częścią teoretyczną do kolejnego ćwiczenia.
3. Wykonać zadania pomocnicze do kolejnego ćwiczenia.