

Instytut Informatyki Stosowanej

Automatyzacja Obliczeń Inżynierskich

Laboratorium

Ćwiczenie 3.

Praca i programowanie w środowisku MATLAB.

Opracował: dr hab. inż. Jacek Kucharski

dr inż. Piotr Urbanek

Program ćwiczenia

OBLICZENIA W PRZESTRZENI ROBOCZEJ ŚRODOWISKA MATLAB.

DZIAŁANIA NA MACIERZACH

Uwaga! Przed przystąpieniem do wykonania zadań zalecane jest utworzenie dziennika wykonywanych poleceń w przestrzeni roboczej. Ułatwi to później wykonanie sprawozdania z laboratorium. Składnia funkcji inicjującej działanie dziennika jest następująca:

`diary [nazwa pliku.txt] %rozpoczęcie trybu przekazywania strumienia danych z klawiatury do pliku`

polecenia wykonywane w przestrzeni roboczej.....

`diary off %Wyłączenie trybu dziennika`

Zadania do wykonania:

1. Utworzyć na możliwie wiele sposobów macierze liczb rzeczywistych postaci:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

2. Wykorzystując macierz A utworzyć macierz liczb zespolonych postaci:

$$B = \begin{bmatrix} 1+2i & 2+3i & 3+4i \\ 4+5i & 5+6i & 6+7i \\ 7+8i & 8+9i & 9+10i \end{bmatrix}$$

3. Wykonać podstawowe działania macierzowe na macierzach A i B tj.: $[A+B]$, $[A-B]$, $[A*B]$, $[A/B]$, $[A\backslash B]$, $[A']$, $[B']$, $[A^2]$, $[B^2]$.
4. Wykonać podstawowe działania tablicowe na macierzach A i B tj.: $[A.*B]$, $[A./B]$, $[A.\backslash B]$, $[A.^2]$, $[B.^2]$.
5. Na podstawie macierzy A i B stworzyć macierze C i D postaci:

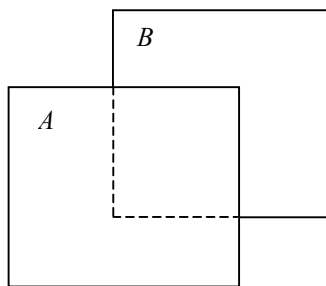
$$C = \begin{bmatrix} -1 & 2 & 3 \\ -4 & 5 & 6 \\ -7 & 8 & 9 \end{bmatrix}; \quad D = \begin{bmatrix} 1+2i & 2+3i & 3+4i \\ 4+5i & 5+6i & 6+7i \\ -7-8i & -8-9i & -9-10i \end{bmatrix}$$

6. Obliczyć części rzeczywiste i urojone oraz moduły i argumenty elementów macierzy C i D , korzystając odpowiednio z funkcji `real`, `imag`, `abs`, `angle`. Porównać i skomentować uzyskane wyniki.
7. Porównać sposoby obliczania macierzy sprzężonej i transponowanej macierzy D wykorzystując operatory `'` oraz `.'`.

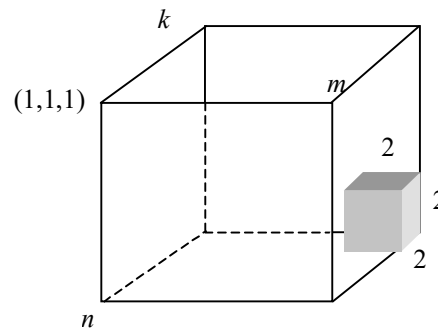
TABLICE WIELOWYMIAROWE, KOMÓRKOWE I STRUKTURALNE

8. Wykorzystując polecenie `cat` zbudować z macierzy A i B trójwymiarową tablicę AB , tak jak to pokazano na rys.6a.

a)



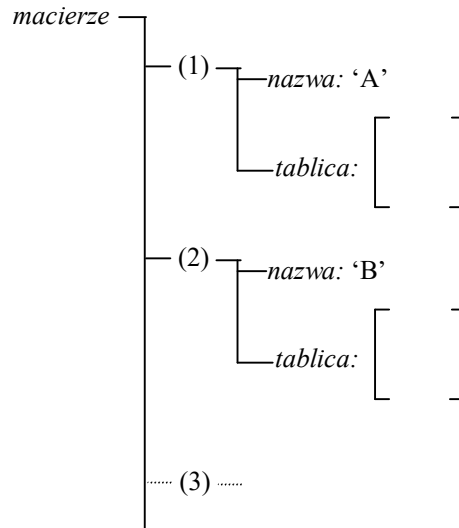
b)



Rys.6.

9. Napisać wyrażenia, w formie niezależnej od faktycznych wymiarów m, n, k tablicy trójwymiarowej (patrz rys.6b), realizujące następujące czynności:
 - obliczanie różnicy odpowiadających sobie elementów górnej i dolnej „powierzchni” tablicy trójwymiarowej;
 - utworzenie macierzy o wymiarach $n \times 4$ składającej się ze wszystkich pionowych „krawędzi” tablicy trójwymiarowej;
 - podwojenie wartości elementów tablicy zwartych w „obszarze” $2 \times 2 \times 2$ wskazanym na rys.6b.
 UWAGA: każdy podpunkt zadania należy zrealizować w postaci jednego polecenia, którego poprawność należy sprawdzić w odniesieniu do utworzonej wcześniej tablicy AB .
10. Utworzyć tablicę komórkową c_{AB} o wymiarach 3×2 , w której elementami kolumn będą: w pierwszej nazwy tablic (A , B , AB), a w drugiej – odpowiadające nazwom tablice.

11. Wykorzystując elementy tablicy komórkowej *c_AB* obliczyć sumę macierzy *A* i *B*, umieszczając wynik w tablicy *c_AB* jako nowy element (np. w czwartym wierszu pierwszej kolumny).
12. Utworzyć strukturę o nazwie *macierze*, w której zawarte zostaną macierze *A* i *B* zgodnie ze schematem podanym na rys.7.



Rys.7

13. Wykorzystując pola struktury *macierze* obliczyć sumę macierzy *A* i *B* umieszczając wynik w strukturze *macierze* jako nowy element.
14. Obliczyć sumę wybranego elementu struktury *macierze* i wybranej komórki tablicy *c_AB*.

Opracowanie sprawozdania

Zapisać fragmenty sesji pracy z programem obejmujące ostateczne rozwiązanie poszczególnych punktów instrukcji.

PROGRAMOWANIE W ŚRODOWISKU MATLAB

1. Napisać m-pliki funkcyjne realizujące za pomocą wielkości skalarnych (iteracyjnie) wybrane jedno- i dwuargumentowe operacje macierzowe i tablicowe ([+], [*], ['], [.'], [^], [.^]). Funkcje powinny sprawdzać rozmiary argumentów i informować o ewentualnych nieprawidłowościach. Należy także uwzględnić możliwość występowania skalarów.

2. Zbudować m-plik skryptowy będący nadrzędnym programem dla stworzonych w pkt. 1 m-plików funkcyjnych. Skrypt powinien umożliwiać:
 - wprowadzanie danych (argumentów) w wierszu poleceń - np. funkcja input, inputdlg,
 - wybór wykonywanej operacji - np. funkcja menu,
 - sprawdzenie poprawności wykonywanych przez m-pliki funkcyjne operacji wykorzystując wbudowane operatory macierzowe i tablicowe,
 - porównanie czasochłonności operacji realizowane za pomocą m-pliku i operatora wbudowanego.

3. Napisać dwa m-pliki funkcyjne ze zmienną liczbą argumentów wejściowych i wyjściowych:
 - plik obliczający sumę lub iloczyn dowolnej liczby argumentów, przy czym jako pierwszy parametr wejściowy należy uwzględnić możliwość podawania (w postaci odpowiedniego symbolu) rodzaju wymaganej operacji ([+], [*], [.*]); program powinien sprawdzać rozmiary wprowadzanych argumentów, odrzucając te, które nie spełniają odpowiednich wymagań,
 - plik wykonujący transpozycję nieokreślonej z góry liczby macierzy, przy czym w przypadku argumentów zespolonych należy dla każdego z takich argumentów poprosić użytkownika o podanie rodzaju transpozycji.