



**POLITECHNIKA
GDAŃSKA**

**WYDZIAŁ ELEKTROTECHNIKI I AUTOMATYKI
Katedra Inżynierii Systemów Sterowania**

PODSTAWY AUTOMATYKI

**MATLAB - komputerowe środowisko obliczeń naukowo-
inżynierskich - podstawowe operacje na liczbach
i macierzach.**

***Materiały pomocnicze do ćwiczeń laboratoryjnych -
część I - termin T1***

Opracowanie:

Kazimierz Duzinkiewicz, dr hab. inż.

Michał Grochowski, dr inż.

Robert Piotrowski, dr inż.

Tomasz Rutkowski, dr inż.

Rafał Łangowski, dr inż.

Gdańsk

1. MATLAB - ogólna charakterystyka środowiska

MATLAB jest interakcyjnym środowiskiem programowym przeznaczonym do wykonywania różnorodnych obliczeń numerycznych i graficznego przedstawiania danych i wyników obliczeń. Nazwa MATLAB pochodzi od słów **MAT**rix **LAB**oratory. Jak sugeruje nazwa, podstawowym typem danych używanych w MATLAB'ie są **macierze** (rzeczywiste lub zespolone). Środowisko to tworzą:

- **Interpreter języka programowania** MATLAB'a wraz z bibliotekami podstawowych działań i obliczeń na macierzach (odwracanie macierzy, rozkłady macierzy, wartości własne, itp.).
- **Standardowe biblioteki procedur** napisanych w języku MATLAB'a realizujące różnorodne obliczenia; należą do nich m. in.: procedury realizujące przekształcenia macierzy, obliczanie wartości funkcji elementarnych i specjalnych, całkowanie numeryczne, rozwiązywanie układów równań różniczkowych zwyczajnych, podstawowe obliczenia statystyczne.
- **Dodatkowe biblioteki procedur** napisanych w języku MATLAB'a (ang. *toolbox*; tłumaczenie polskie: przybornik lub skrzynka narzędziowa), które nabywa się oddzielnie w zależności od potrzeb użytkownika. Dostępne są, m. in. przyborniki: cyfrowego przetwarzania sygnałów - *Signal Processing Toolbox*, identyfikacji systemów - *System Identification Toolbox*, projektowania i analizy liniowych układów sterowania - *Control System Toolbox*, projektowania i analizy krzepkich układów sterowania - *Robust Control Toolbox*, analizy i projektowania nieliniowych układów sterowania - *Nonlinear Control Design Toolbox*, optymalizacji - *Optimization Toolbox*, obliczeń statystycznych - *Statistics Toolbox*, analizy i projektowania sieci neuronowych - *Neural Network Toolbox*, logiki rozmytej - *Fuzzy Logic Toolbox*.
- **Nakładki** napisane w języku MATLAB'a, które są aplikacjami ułatwiającymi realizację obliczeń określonego rodzaju. Przykładem nakładki jest SIMULINK - nakładka umożliwiająca graficzne definiowanie struktury układu dynamicznego oraz przeprowadzanie symulacji na tym układzie.

2. Uruchomienie

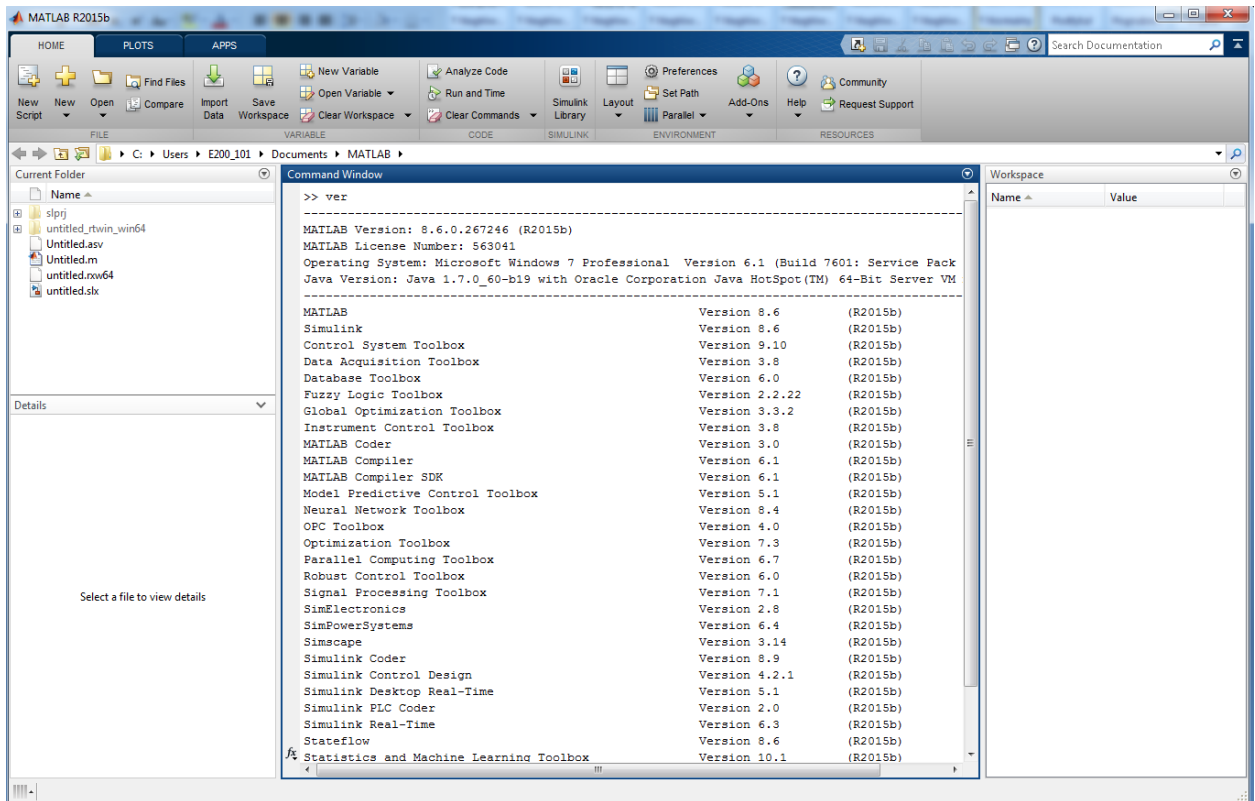
MATLAB jest programem pracującym w środowisku WINDOWS, można go zatem uruchomić będąc w tym środowisku. Uruchomienia można dokonać dowolnym ze sposobów właściwych dla tego środowiska.

W opracowaniu dalszej części materiału korzystano ze środowiska Matlab w wersji R2015b (8.6.0.267246).

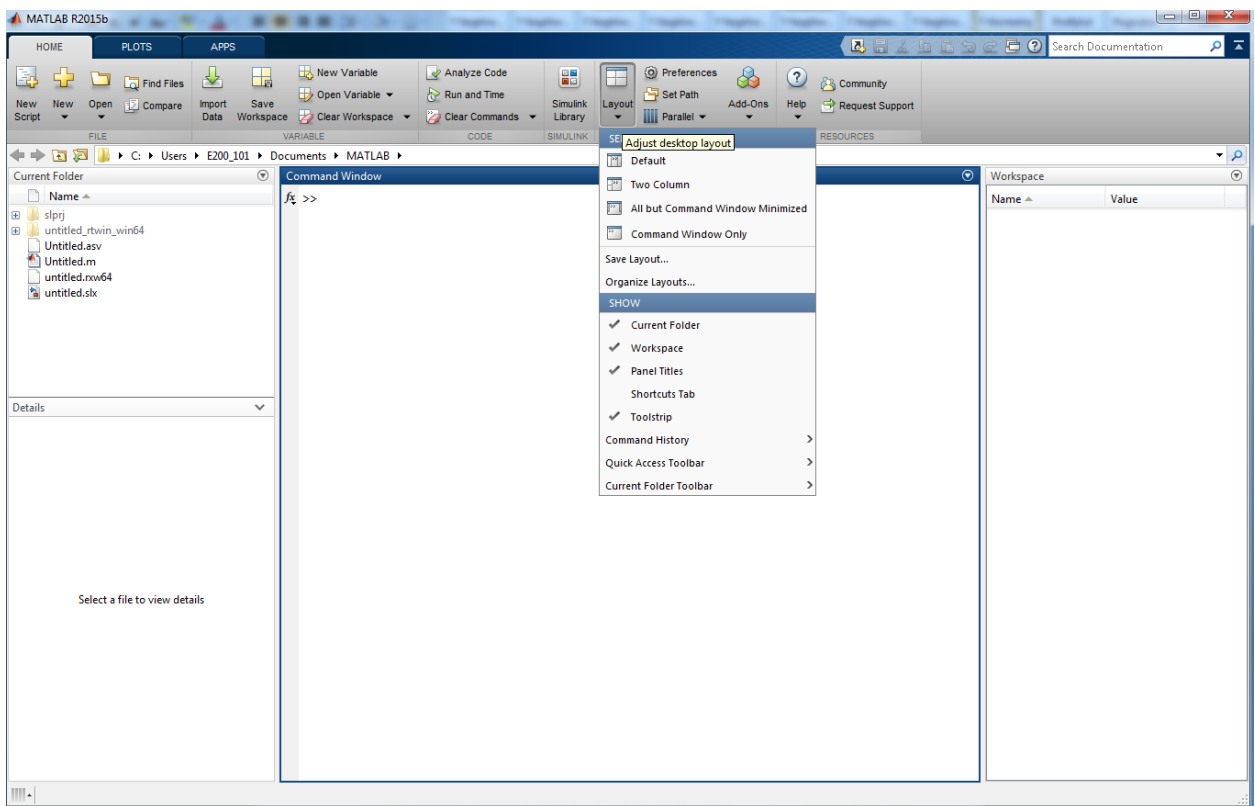
Po uruchomieniu powinno pojawić się główne okno MATLAB'a: okno poleceń **MATLAB Command Window**. W tym oknie, powinien pojawić się znak zachęty (prompt): **>>** oraz migający kursor (pionowa kreska). Szczegółowe informacje na temat zainstalowanej wersji i składników środowiska MATLAB uzyskamy wpisując polecenie (rys. 1):

» **ver.**

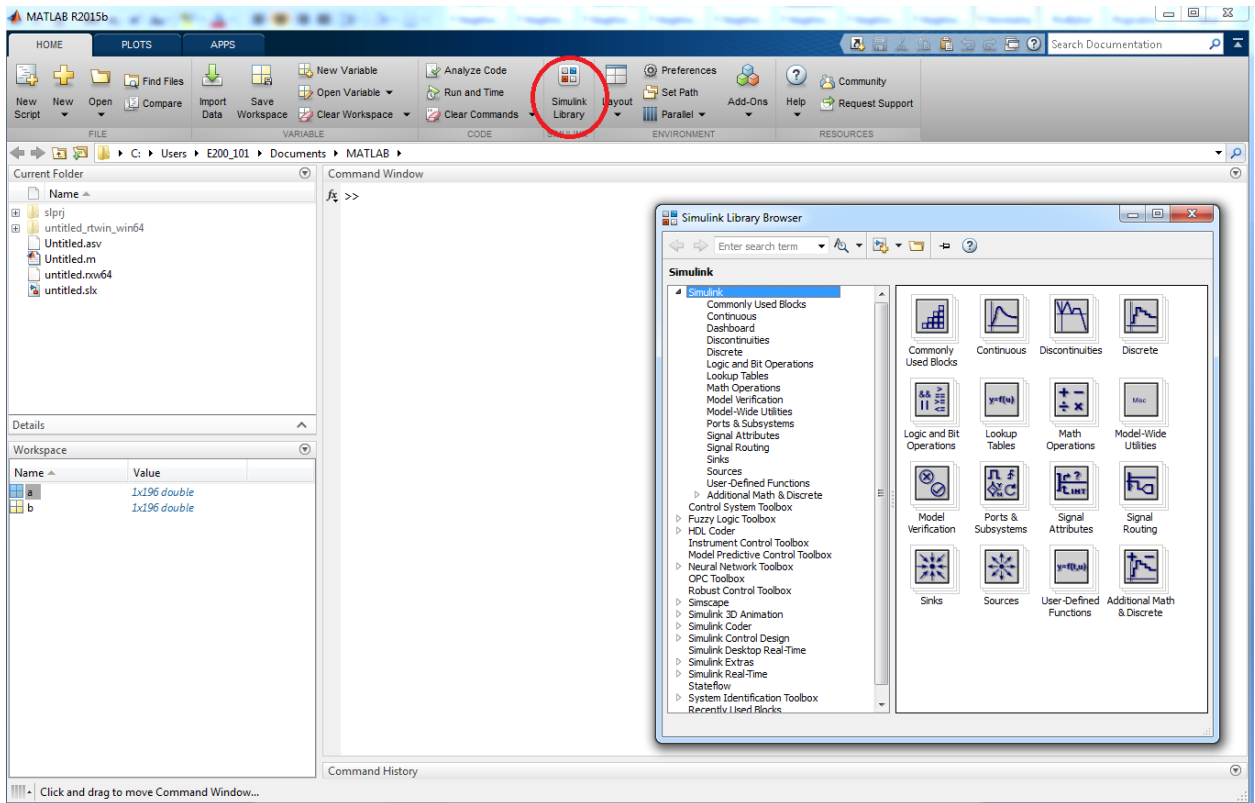
Na rys. 1 przedstawiono domyślny (standardowy) widok ustawień okien MATLAB'a. Widok ten można zmieniać zgodnie z własnymi preferencjami, korzystając z polecenia **layout** (rys. 2). Ponadto, w pasku narzędziowym znajdują się trzy zakładki, odpowiednio: **Home**, **Plots** oraz **Apps**. Pierwsza z nich zawiera podstawowe polecenia oraz umożliwia wywołanie aplikacji SIMULINKa (rys. 3). Druga umożliwia szybkie tworzenie różnorodnych reprezentacji graficznych danych zmiennych (rys. 4). Natomiast zakładka Apps zawiera szereg nakładek dla różnych funkcjonalności dostępnych w poszczególnych przybornikach (rys. 5).



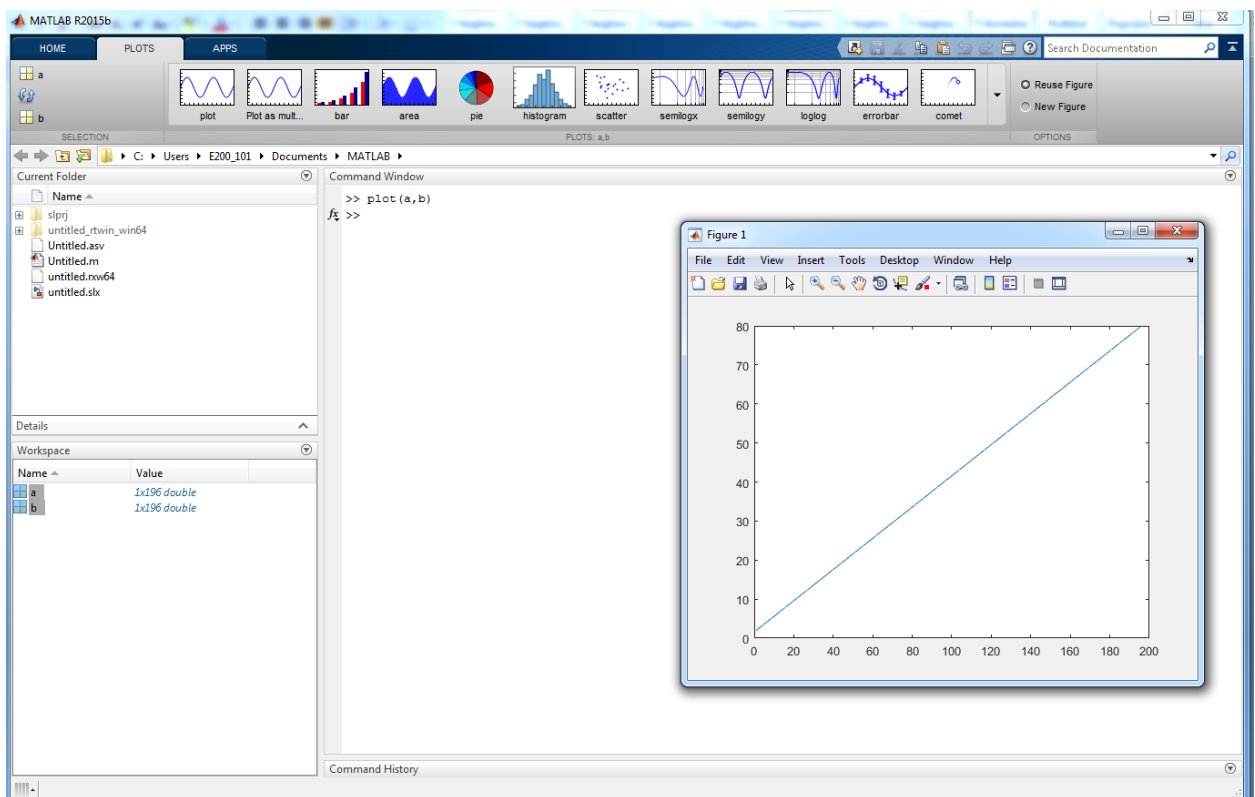
Rys. 1. Widok ogólny środowiska MATLAB



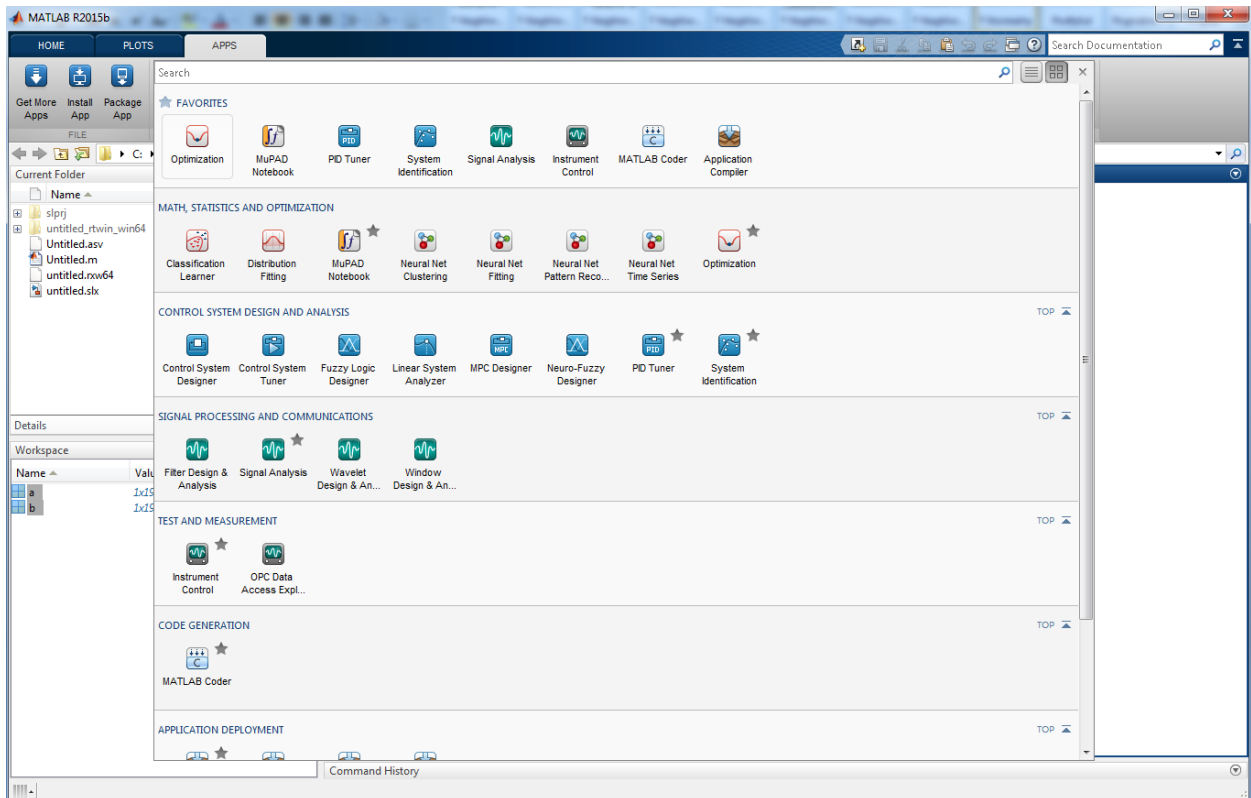
Rys. 2. Widok ogólny środowiska MATLAB (polecenie: layout)



Rys. 3. Widok ogólny środowiska MATLAB (pasek narzędziowy - zakładka Home)



Rys. 4. Widok ogólny środowiska MATLAB (pasek narzędziowy - zakładka Plots)



Rys. 5. Widok ogólny środowiska MATLAB (pasek narzędziowy - zakładka Apps)

3. Praca w środowisku MATLAB'a

Znak zachęty oznacza, że użytkownik może wprowadzać polecenia MATLAB'a. MATLAB stanowi w istocie interpreter języka, zaprojektowanego specjalnie z myślą o obliczeniach numerycznych i graficznej wizualizacji danych i wyników obliczeń.

Polecenia wprowadza się z klawiatury, w oknie Command Window. W istocie praca w środowisku MATLAB'a polega na wydawaniu poleceń, które po zatwierdzeniu przez użytkownika naciśnięciem klawisza **Enter** są wykonywane przez interpreter. W ten sposób bezpośrednio z wiersza poleceń można:

- zdefiniować zmienną MATLAB'a,
- wywołać funkcję MATLAB'a,
- wywołać procedurę MATLAB'a zbudowaną z poleceń interpretera i zapisaną w pliku tekstowym o nazwie *.m (pliki te nazywane są **M-plikami** lub skryptami),
- wydawać polecenia przekazywane do wykonania w systemie operacyjnym DOS lub środowisku WINDOWS.

Gdy tekst polecenia nie mieści się w jednej linii, można przejść do następnej linii wpisując wielokropek (...) przed naciśnięciem klawisza Enter. Każde polecenie, które nie zostanie zakończone średnikiem (;) powoduje wyświetlenie odpowiedzi.

W środowisku MATLAB'a rozróżniane są duże i małe litery. Wyjątkiem są nazwy M-plików i poleceń systemu operacyjnego DOS, które też mogą być wydawane ze środowiska MATLAB'a. W przypadku pracy w środowisku WINDOWS, możemy wykorzystać wszystkie mechanizmy i narzędzia tego środowiska.

4. Zakończenie pracy z MATLAB'em

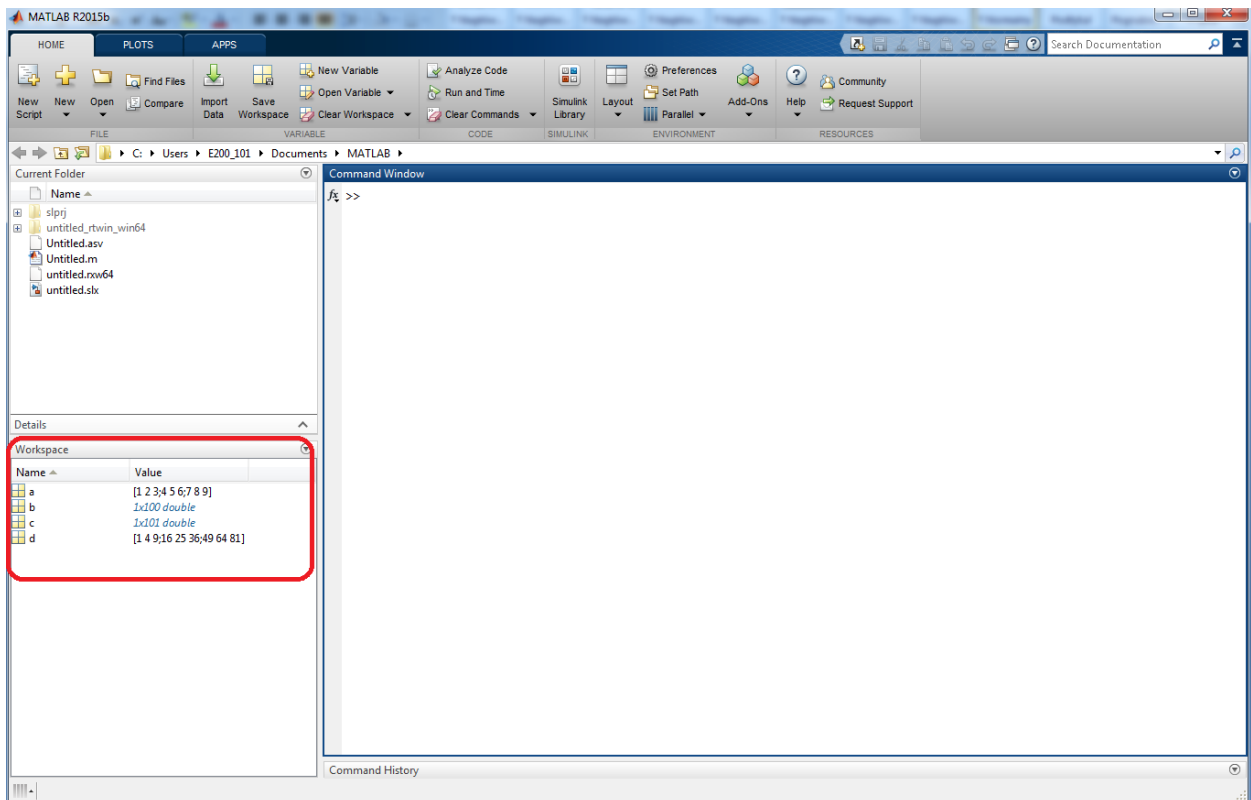
Zakończenie pracy w środowisku MATLAB'a następuje przez wykonanie polecenia:

`>> exit` lub równoważnego `>> quit`

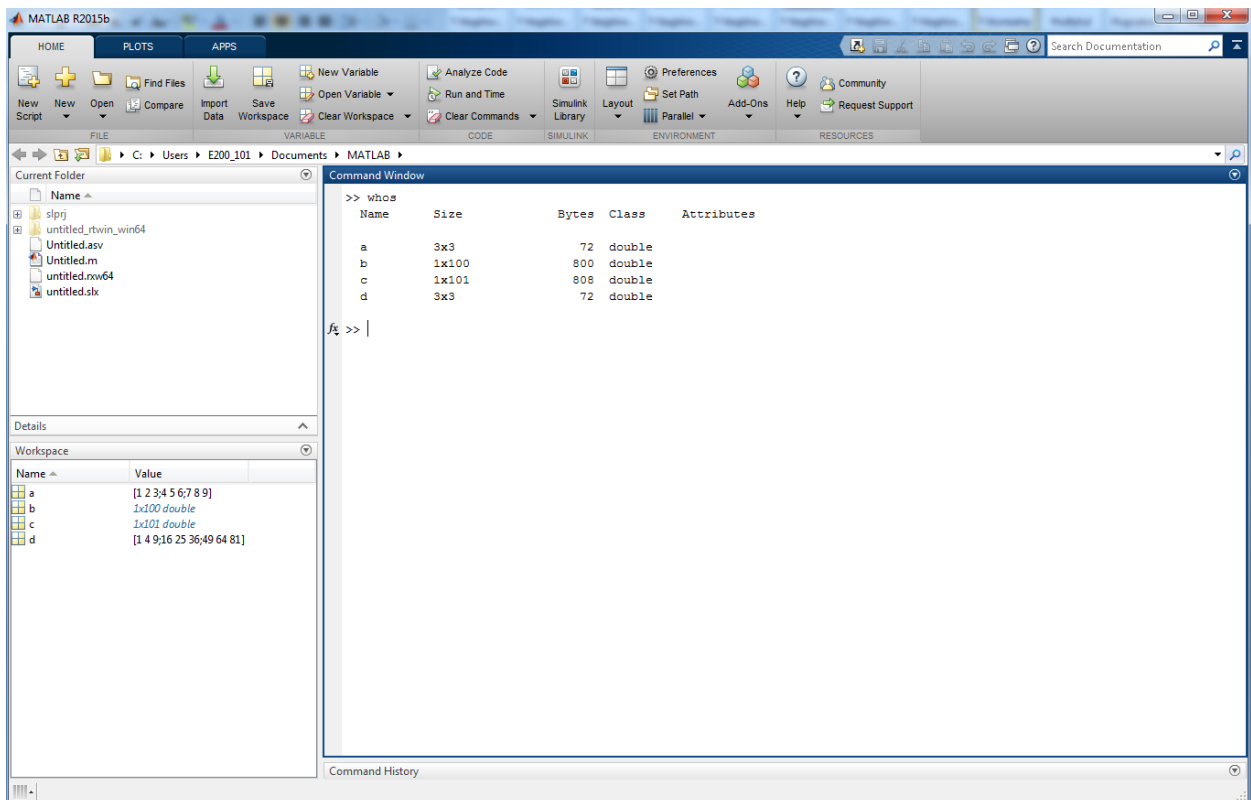
Pracując w środowisku WINDOWS możemy też zakończyć pracę z MATLAB'em wykorzystując jedną z opcji zamykania aplikacji pracujących w środowisku WINDOWS.

5. Przestrzeń robocza MATLAB'a

Wszystkie zmienne tworzone w wyrażeniach są przechowywane w **przestrzeni roboczej** MATLAB'a (**workspace**). Zawartość tej pamięci możemy odczytać bezpośrednio w zakładce Workspace (rys. 6) lub w oknie Command Window (rys. 7) wpisując polecenie: » **whos**.



Rys. 6. Przestrzeń robocza środowiska MATLAB



Rys. 7. Wyświetlenie zawartości przestrzeni roboczej (polecenie: whos)

Poza zmiennymi nazwanymi przez użytkownika w przestrzeni roboczej przechowywane są dwie zmienne o nazwach nie nadawanych przez użytkownika, są to **ans** i **eps**. Zmienna **ans** opisana została w pkt. 6. Natomiast zmienna **eps** jest tolerancją dla określenia wielkości bliskich zeru oraz określa dokładność przedstawiania liczb zmiennoprzecinkowych. Jej wartość jest związana z arytmetyką komputera na którym pracujemy w środowisku MATLAB'a. Dla komputerów standardu PC wynosi ona domyślnie zwykle:

$$eps = 2^{-52}$$

co odpowiada w przybliżeniu:

$$eps \cong 2,2204 \cdot 10^{-16}$$

Oznacza to, że 15 i 16 cyfra liczby może być niepoprawna.

Zmienne przechowywane w przestrzeni roboczej można z niej usuwać. Służy do tego polecenie **clear**. Wybrane postacie polecenia **clear** są następujące:

```
clear
clear name
clear name1 name2 ...
```

Polecenie **clear** usuwa z przestrzeni roboczej wszystkie zmienne. Pozostawia ono przestrzeń roboczą pustą, tak jak byśmy przed chwilą przywołali MATLAB'a. Polecenie **clear name** lub **clear name1 name2 ...** usuwa z przestrzeni roboczej zmienne o podanych nazwach.

Przykłady: niech w przestrzeni roboczej będą następujące zmienne:

» **who**

Your variables are:

```
A    b    d    f    h
a    c    e    g
»
```

» **clear a**

» **who**

Your variables are:

```
A    c    e    g
b    d    f    h
»
```

» **clear b c**

» **who**

Your variables are:

```
A    e    g
d    f    h
»
```

Informacje wyświetlane w Command Window mogą być wyczyszczone poleceniem **clc**.

Zakończenie sesji MATLAB'a powoduje utratę wszystkich zmiennych przechowywanych w przestrzeni roboczej. Przed zakończeniem sesji MATLAB'a cała zawartość przestrzeni roboczej może być zapamiętana za pomocą polecenia **save**.

» **save**

Saving to: matlab.mat

»

Jak podaje odpowiedź MATLAB'a zawartość przestrzeni roboczej została zapamiętana w pliku o nazwie matlab.mat. Użycie polecenia **load** na początku następnej sesji spowoduje załadowanie całej zawartości uprzednio zapamiętanej przestrzeni roboczej.

» **load**

Loading from: matlab.mat

»

Można oczywiście zapamiętywać wybrane fragmenty (wybrane zmienne) aktualnej zawartości przestrzeni roboczej. Wybrane postacie polecenia **save** są następujące:

save file

save file *name1 name2 ...*

Polecenie **save file** zapisuje całą zawartość przestrzeni roboczej w pliku o nazwie file.mat. Polecenie **save file name1 name2 ...** zapisuje zmienne o nazwach **name1 name2 ...** w pliku o nazwie file.mat. Przykłady działania tych poleceń:

Przykład 1:

» who

Your variables are:

A b d f h

a c e g

» save mywork1

» clear

» who

Your variables are:

» load mywork1

» who

Your variables are:

A b d f h

a c e g

»

Przykład 2:

» save mywork2 A a b c

» save mywork3 d e f g h

» clear

» who

Your variables are:

» load mywork2

» who

Your variables are:

A a b c

» load mywork3

» who

Your variables are:

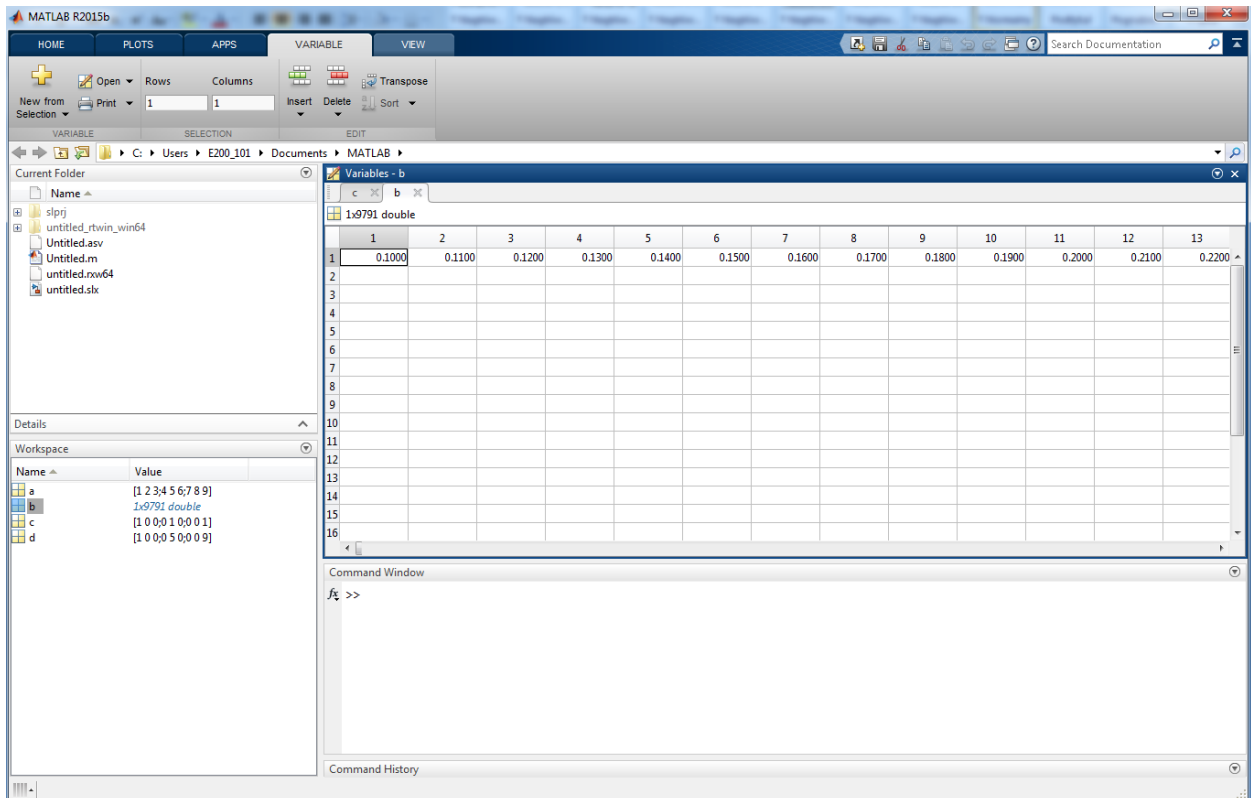
A b d f h

a c e g

»

Opisane wcześniej polecenia (jak i wiele innych) można wykonać z poziomu zakładki Home paska narzędziowego, np. Save Workspace, Open File, itp.

Wszystkie wartości zmiennych zapisane w przestrzeni roboczej można wyświetlić poprzez wpisanie w Command Window ich nazw lub dwukrotne kliknięcie na daną zmienną w zakładce workspace. W efekcie otworzy się tzw. **Array Editor**, który pozwala na podgląd wartości ukrytych pod nazwą danej zmiennej oraz ich edycję (rys. 8).



Rys. 8. Wyświetlenie zawartości poszczególnych zmiennych (Array Editor)

6. Wyrażenia i zmienne

MATLAB jest językiem zmiennych i wyrażeń. Zmienne i wyrażenia pojawiają się w poleceniach formułowanych przez użytkownika. Polecenie MATLAB'a ma najczęściej formę:

zmienna = wyrażenie

lub prościej:

wyrażenie

Wyrażenia są budowane ze zmiennych, operatorów i innych znaków specjalnych, funkcji. Obliczona po zaakceptowaniu polecenia wartość wyrażenia przypisana zostaje zmiennej dla ewentualnego późniejszego wykorzystania. Wartość ta w formie odpowiedzi MATLAB'a może być wyświetlona w oknie poleceń. Jeżeli nazwa zmiennej i znak = nie wystąpią w poleceniu tworzona jest automatycznie (robocza) zmienna o nazwie **ans** i jej przypisywana jest wartość wyrażenia. Na przykład napisanie polecenia:

» 1990/81

spowoduje wyświetlenie w oknie poleceń odpowiedzi:

```
ans =
    24.5679
»
```

Polecenie jest zwykle kończone naciśnięciem klawisza **Enter**. Jeżeli jednak ostatnim znakiem polecenia jest **;**, odpowiedź nie pojawia się w oknie poleceń, chociaż przypisanie wartości następuje. Jest to szczególnie użyteczne przy pisaniu M-plików oraz wtedy, gdy wynik polecenia jest na tyle obszerny, że

znajomość poszczególnych liczb nie jest istotna. Na przykład polecenie mnożenia dwóch wielomianów wyrażonych przez wektory ich współczynników:

» `c=conv(a,b);`

»

spowoduje przemnożenie odpowiednich składowych wektorów *a* i *b* oraz ich dodanie, dokona przypisania wyniku zmiennej *c*, ale wynik nie zostanie podany w oknie poleceń.

Jeżeli wyrażenie jest na tyle skomplikowane, że polecenie nie mieści się w jednej linii wówczas można zastosować przeniesienie przez napisanie trzech lub więcej kropek przed naciśnięciem klawisza **Enter**. Pamiętać przy tym należy, aby pomiędzy ostatnim znakiem wyrażenia, które chcemy przenieść a znakami przeniesienia postawić separator - spację. Na przykład:

» `s=1-1/2+1/3-1/4+1/5-1/6+1/7 ...`
`-1/8+1/9-1/10+1/11-1/12`

s =

0.6532

»

7. Bibliografia

Brzózka J. *Ćwiczenia z automatyki w Matlabie i Simulinku*. Wydawnictwo MIKOM, 1997.

Brzózka J., Dorobczyński L. *Matlab - środowisko obliczeń naukowo - technicznych*. Wydawnictwo MIKOM, 2005.

Mrozek B., Mrozek Z. *Matlab i Simulink. Poradnik użytkownika. Wydanie II*. Wydawnictwo HELION, 2004.

Pratap R. *Matlab 7 dla naukowców i inżynierów*. Wydawnictwo HELION, 2010.

Sradomski W. *Matlab. Praktyczny poradnik modelowania*. Wydawnictwo HELION, 2015.

Zalewski A., Cegieła R. *Matlab - obliczenia numeryczne i ich zastosowania*. Wydawnictwo NAKOM, 1996.