

2. Pierwsza sesja z MATLAB-em

Podstawowym oknem MATLAB-a jest okno **Command Window**. W oknie tym można wpisywać polecenia lub uruchamiać napisane przez użytkownika skrypty. W MATLAB-ie można korzystać z prostych poleceń wykonujących obliczenia arytmetyczne, można również wykonywać skomplikowane analizy statystyczne itp.

2.1. Wpisywanie poleceń

Na początku spróbujemy wykonać proste obliczenia np.: $2+2$ czy $4*5$. W tym celu należy wpisać w oknie **Command Window** po znaku zachęty (\gg) odpowiednie polecenie a następnie nacisnąć klawisz **Enter** np.:

```
>> 2+2
```

Po wykonaniu polecenia, pod jego tekstem, zostanie wyświetlony wynik:

```
ans =  
    4
```

jednocześnie znowu pojawi się wolna linia ze znakiem zachęty, w której użytkownik może wpisać kolejne polecenie np.:

```
>> 4*5
```

Po naciśnięciu klawisza **Enter**, zostanie wyświetlony wynik:

```
ans =  
    20
```

Wyniki wszystkich obliczeń, o ile użytkownik tego nie zmieni, przechowywane są w zmiennej o nazwie **ans** (skrót od ang. answer – odpowiedź). Po wykonaniu pierwszego polecenia ($2+2$) zmienna **ans** miała wartość 4. Zmienna przechowuje swoją wartość tak długo, dopóki jakieś nowe polecenie nie zmieni jej wartości. Po wykonaniu drugiego polecenia zmienna **ans** otrzymała nową wartość tzn. 20.

Użytkownik może zażądać aby obliczona wartość była przechowana w innej zmiennej niż zmienna **ans**. W tym celu musi wpisać polecenie o składni:

```
zmienna = wyrażenie lub zmienna = wartość.
```

W podanym powyżej zapisie „zmienna” jest wymyśloną przez użytkownika nazwą.

Uwaga!

W nazwach zmiennych można używać jedynie dużych i małych liter alfabetu (zmienna odp i zmienna ODP to dwie różne zmienne), cyfr oraz znaku podkreślnika (_). Dodatkowo, nazwa nie może rozpoczynać się od cyfry, nie są również dozwolone polskie znaki: ą, ć, ę itd.



Załóżmy, że użytkownik chce zapamiętać w zmiennej o nazwie **x** wartość wyrażenia $2+2$, a w zmiennej o nazwie **y** wartość 3. W tym celu musi wykonać kolejno dwa polecenia. Poniżej przedstawiono wygląd okna **Command Window** po wykonaniu obu poleceń:

```
>> x = 2+2
```

```
x =  
    4
```

```
>> y = 3
```

```
y =  
    3
```

Wartości zapamiętane w utworzonych przez użytkownika zmiennych mogą być następnie wykorzystane w kolejnych obliczeniach:

```
>> z = 2*(x+y)
```

```
z =  
   14
```

W wyrażeniach użytkownik może używać różnych predefiniowanych funkcji np.: trygonometrycznych czy statystycznych. Podstawowe funkcje zostaną omówione w punkcie następnym. W przedstawionym poniżej przykładzie obliczana jest wartość cosinusa kąta 0 radianów, wynik zapamiętywany jest w zmiennej **q**:

```
>> q = cos(0)
```

```
q =  
    1
```

W celu wyświetlenia wartości jednej z wcześniej zdefiniowanych zmiennych wystarczy jako polecenie wpisać nazwę tej zmiennej np.:

```
>> x
```

```
x =  
    4
```

2.2. Błędy i ich poprawianie

Załóżmy, że użytkownik pomylił się wpisując polecenie, użył operatora który nie istnieje (#):

```
>> a = 2 # 2
```

W takiej sytuacji MATLAB zareaguje komunikatem o błędzie:

```
??? 2      # 2  
      |
```

Error: Missing operator, comma, or semicolon.

Komunikaty wyświetlane są zawsze czerwoną czcionką. Tekst komunikatu zależy od popełnionego błędu. W przedstawionej powyżej sytuacji wyświetlana jest informacja o zgubionym operatorze, przecinku lub średniku. Dodatkowo linią pionową zaznaczone jest miejsce pojawienia się błędu.

Innym często popełnianym błędem jest błąd podczas wpisywania nazwy funkcji. Załóżmy, że użytkownik chce policzyć wartość cosinusa dla kąta 0 radianów:

```
>> b = cs(0)
```

Błąd w tym przypadku wynika ze źle wpisanej nazwy funkcji: powinno być **cos** jest **cs**. MATLAB zareaguje komunikatem:

```
??? Undefined function or variable 'cs'.
```

W tłumaczeniu oznacza to: **Nieokreślona funkcja lub zmienna 'cs'**.

Istnieje wiele różnych metod pozwalających poprawić popełnione błędy, np.:



- można wpisać ponownie polecenie, tym razem bez błędu
- można skorzystać z klawiszy ↑ oraz ↓. Po naciśnięciu ↑ MATLAB wyświetli w linii poleceń (po znaku zachęty >>) poprzednie polecenie, które będzie można edytować (ustawić się w miejscu błędu, cos dopisać, coś usunąć). Kilkakrotne naciskanie klawisza ↑ powoduje wyświetlanie poleceń wykonywanych wcześniej, klawisz ↓ pozwala na powrót do późniejszych komend.
- można skorzystać z okna **Command History** (zostanie omówione w podpunkcie następnym)

2.3. Okno Command History

Okno historii komend spełnia dwie podstawowe funkcje:






- pozwala na przeglądanie wykonywanych w poleceń. Widoczne w oknie polecenia poprzedzane są informacją o dacie i godzinie rozpoczęcia pracy z MATLAB-em.



- umożliwia pobranie uruchomionego wcześniej polecenia do ponownego wykonania, np. w przypadku popełnienia błędu w trakcie wpisywania polecenia można je pobrać z okna „historii” do okna „komend”, poprawić i uruchomić ponownie. W celu pobrania polecenia można:
 - przeciągnąć myszą polecenie z okna „historii” do okna „komend”,
 - wykorzystać mechanizm kopiowania i wklejania. W oknie historii należy wybrać określone polecenie, wykonać kopiowanie (polecenie **Copy** z menu **Edit** lub z menu podręcznego (prawy przycisk myszy), można również wykorzystać przycisk znajdujący się na pasku narzędzi: ) , następnie polecenie należy wkleić do okna komend (polecenie **Paste** z menu **Edit** lub z menu podręcznego lub przycisk ).

2.4. Okno Workspace

Okno pozwala przeglądać zmienne utworzone podczas bieżącej sesji pracy z MATLAB-em. Po wykonaniu poleceń z podpunktu „Wpisywanie poleceń” okno **Workspace** zawiera listę utworzonych przez użytkownika zmiennych.

Name	Size	Bytes	Class
 y	1x1	8	double array
 z	1x1	8	double array
 x	1x1	8	double array
 ans	1x1	8	double array
 q	1x1	8	double array

Zmienne wyświetlane są kolejnych wierszach listy.

Kolumny listy opisują dokładniej każdą ze zmiennych:

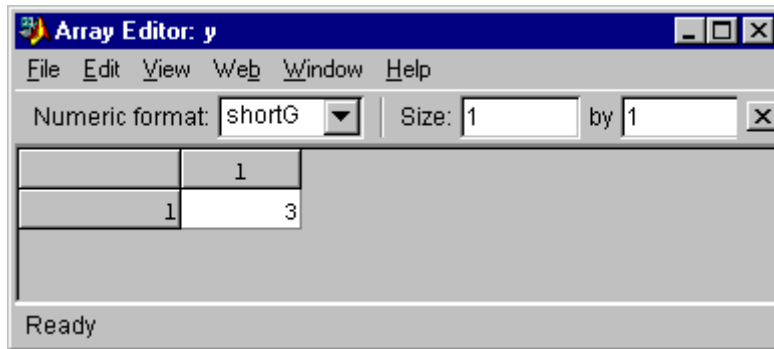
- kolumna **Name** zawiera nazwy zmiennych
- kolumna **Size** – rozmiary zmiennych. MATLAB przechowuje wartości liczbowe w macierzach. Jeżeli w zmiennej **y** jest np. przechowywana liczba 3, to zmienna **y** jest macierzą o jednym wierszu i jednej kolumnie, gdzie wartość 3 jest wpisana do komórki o indeksie 1,1. Wszystkie zmienne utworzone w trakcie pierwszej sesji z MATLAB-em to macierze o wymiarach 1x1
- kolumna **Bytes** – opisuje rozmiar (w bajtach) pamięci potrzebny do przechowania zmiennej
- kolumna **Class** – opisuje typ zmiennej. Wszystkie zmienne to macierze o wartościach rzeczywistych podwójnej precyzji (double).

W liście nie można oglądać wartości zmiennych. Jeżeli użytkownik chce oglądać lub zmodyfikować wartość zmiennej powinien:

- wskazać zmienną w liście
- podwójnie kliknąć myszą.



W wyniku tych operacji wyświetlone zostanie okno:



Na powyższym rysunku pokazane zostało okno edytora dla zmiennej y . Zmienna y jest macierzą o wymiarach 1x1 (Size: 1 by 1). Wartość zmiennej wynosi 3. Użytkownik może zmodyfikować wartość zmiennej (edytując wartość w komórce macierzy), może również zmienić rozmiar macierzy (edytując wartości w linii Size: 1 by 1).

Po zamknięciu okna zmienna y będzie miała postać i wartość nadaną jej przez użytkownika w oknie edytora.

Ćwiczenia

1. Utwórz zmienne: q o wartości 2, x o wartości 5, y o wartości $135 \cdot 234$, z jest sumą x i y , .
2. Oglądaj wartości zmiennych w oknie **Workspace**.
3. Zmień wartość zmiennej q tak aby była macierzą jednostkową o wymiarach 2x2, zmiennej x nadaj wartość 1, zmiennej y wartość 2.
4. Wyświetl wartości zmiennych x , y , z w oknie **Command Window**.
5. Zauważ, że zmienna z nie ma wartości 3 ($x = 1$, $y = 2$, $z = x+y$). Zmienna ta ma wartość wynikającą z poprzednich wartości zmiennych x i y . Wykonaj ponownie polecenie $z = x + y$ – nie wpisuj go – lecz skorzystaj z innych metod opisanych w punkcie „Błędy i ich poprawianie”.