



Capitolul 5. Mediul de programare MATLAB

5. Mediul de programare MATLAB

5.1. Algoritmi

Exemplu Să se determine concentrația procentuală a unei soluții de acid sulfuric obținută prin amestecarea a 50g soluție de concentrație 98% cu 80g soluție de concentrație 10%.



Concentrația procentuală c_F este egală cu raportul dintre cantitatea totală de substanță dizolvată și masa soluției: $c = m_d/m_s * 100$ (%)

Cantitatea totală de soluție obținută în urma amestecării:

$$m_F = m_A + m_B = 50 + 80 = \mathbf{130 \text{ g soluție}}$$

Cantitatea de H_2SO_4 din soluția A:

$$m_{\text{H}_2\text{SO}_4\text{-A}} = c_A * m_A = 0,98 * 50 = 49 \text{ g H}_2\text{SO}_4 \text{ pur}$$

Cantitatea de H_2SO_4 din soluția B:

$$m_{\text{H}_2\text{SO}_4\text{-B}} = c_B * m_B = 0,10 * 80 = 8 \text{ g H}_2\text{SO}_4 \text{ pur}$$

Cantitatea totală de H_2SO_4 obținută în urma amestecării:

$$m_{\text{H}_2\text{SO}_4\text{-F}} = m_{\text{H}_2\text{SO}_4\text{-A}} + m_{\text{H}_2\text{SO}_4\text{-B}} = 49 + 8 = \mathbf{57 \text{ g H}_2\text{SO}_4 \text{ pur}}$$

Concentrația finală c_F a soluției F: $c_F = m_{\text{H}_2\text{SO}_4\text{-F}} / m_F * 100 = 57/130 * 100 = \mathbf{43,85(\%)}$

5. Mediul de programare MATLAB

Etapele parcurse în rezolvare:

1. Atribuirea de simboluri distincte mărimilor cu care se lucrează:
 - Pentru datele inițiale: m_A, c_A, m_B, c_B
 - Pentru valorile intermediare calculate: $m_{H_2SO_4-A}, m_{H_2SO_4-B}, m_{H_2SO_4-F}, m_F$
 - Pentru rezultatul final: c_F
2. Calcularea succesivă a valorilor intermediare și a mărimii finale.
3. Prezentarea rezultatului.

 O succesiune de reguli ce trebuie parcurse după o anumită secvență.

Păstrând această succesiune de reguli și ținând cont de notațiile inițiale se poate realiza rezolvarea altor probleme similare.

OBS. Deoarece majoritatea problemelor care necesita o rezolvare numerica își are originea în fenomene naturale este necesar ca programatorii sa fie atenti ca rezultatele obtinute sa fie concordante cu realitatea (de exemplu nu poate exista o solutie apoasa de acid clorhidric de 90%).

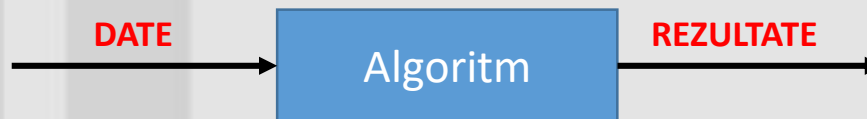
5. Mediul de programare MATLAB

Definiție:

Sucesiunea regulilor care permit rezolvarea unei probleme se numește **ALGORITM**.

Unul dintre primii matematicieni care s-au ocupat de evidențierea regulilor după care se efectuau calculele a fost savantul persan Muhammed Ibn Musa Horezmi cunoscut mai ales sub numele **Al-Horezmi** (~780 ~850 e.n.). Opera sa principală a fost "KITAB AL-DJABR WA'L-MUQUABALA" ("Scurta carte despre calculul prin completare și reducere") a lăsat în urma pe lângă metodele descrise și cuvântul care a desemnat un întreg domeniu matematic: algebra. Numele matematicianului a dat prin adaptare termenul algorithm.

Schematic un algorithm poate fi descris astfel:



5. Mediul de programare MATLAB

Condiții pentru ca o succesiune de evenimente (secvențe, etape, calcule, etc.) să poată fi considerate algoritm:

1. Un algoritm trebuie să fie general.
2. Un algoritm trebuie să aibe un timp finit de calcul (un număr finit de etape).
3. Toate operațiile dintr-un algoritm trebuie să fie definite.
4. Algoritmii trebuie să aibe claritate

Algoritmul de rezolvare a unei probleme nu este unic. Se poate găsi cel puțin încă o metoda pentru rezolvare, chiar dacă această modificare presupune inversarea unor etape.

Nu exista un algoritm de scriere a algoritmilor !

5. Mediul de programare MATLAB

Tipuri de algoritmi

1. În funcție de existența sau inexistența unor reveniri în interiorul algoritmilor se diferențiază:

- algoritmi cu buclă;
- algoritmi fără buclă (liniari).

0 buclă (ciclu) - parcurgerea completă a unui set de operații cu revenire la un punct inițial.

Algoritmii cu buclă se împart în două categorii:

- **algoritmi iterativi**: **Iterativitatea** este procesul prin care rezultatul unei probleme se obține ca urmare a execuției repetate a unui set de operații (repetarea unei bucle sau a unui ciclu), folosind de fiecare dată alte valori de intrare.
- **algoritmi recursivi**: **Recursivitatea** este procesul prin care rezultatul unei probleme se obține ca urmare a execuției repetate a unui set de operații, folosind de fiecare dată rezultatul execuției anterioare.

5. Mediul de programare MATLAB

5.2. Scheme logice

Schema logică - permite descrierea algoritmului.

- este o înșiruire de semne grafice care descriu diversele etape de calcul ale unui algoritm, de la datele inițiale ("citirea" lor) până la rezultate ("afișarea" sau "tipărirea" lor).

O schemă logică are în componență următoarele blocuri:

Bloc de început:



Bloc de sfârșit:



Bloc de citirea datelor:



Bloc de tipărirea rezultatelor:



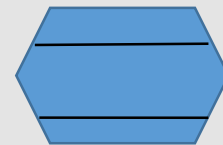
Bloc de calcul:



Bloc de decizie:



Bloc pentru apelul unui subprogram:

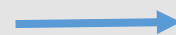


Conector (întreruperea schemei logice):

x – un număr sau o literă



Sensul de parcurgere a schemei:



Obs. Aceste semne nu sunt unice, neexistând un standard universal acceptat.

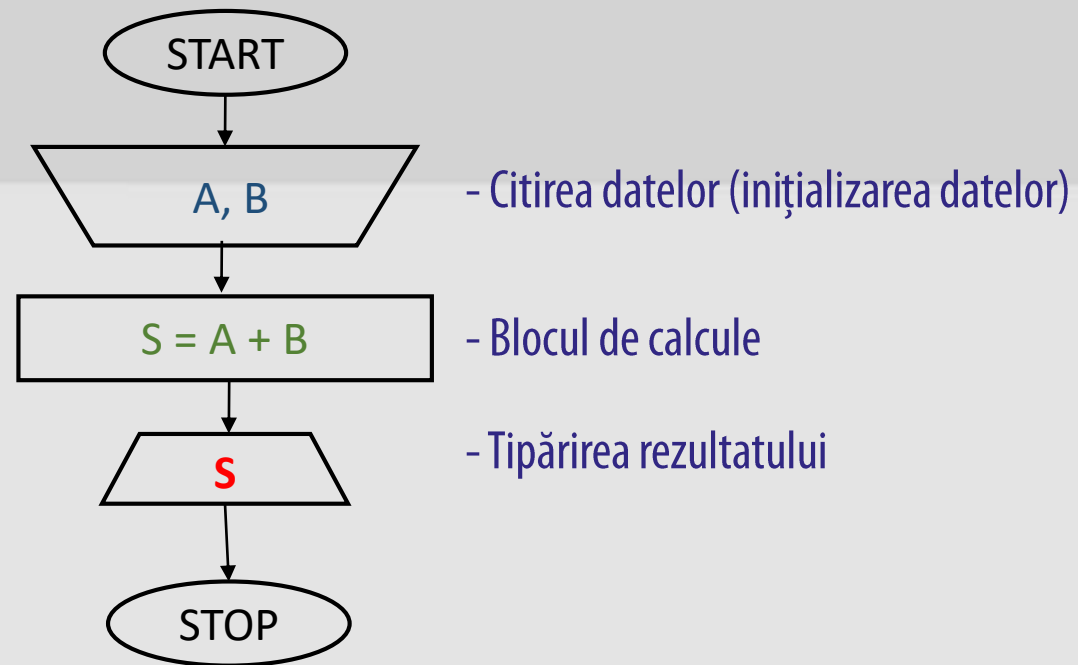
5. Mediul de programare MATLAB

Exemplul 1. Să se construiască schema logică pentru calculul sumei a două numere.

$$S = A + B$$

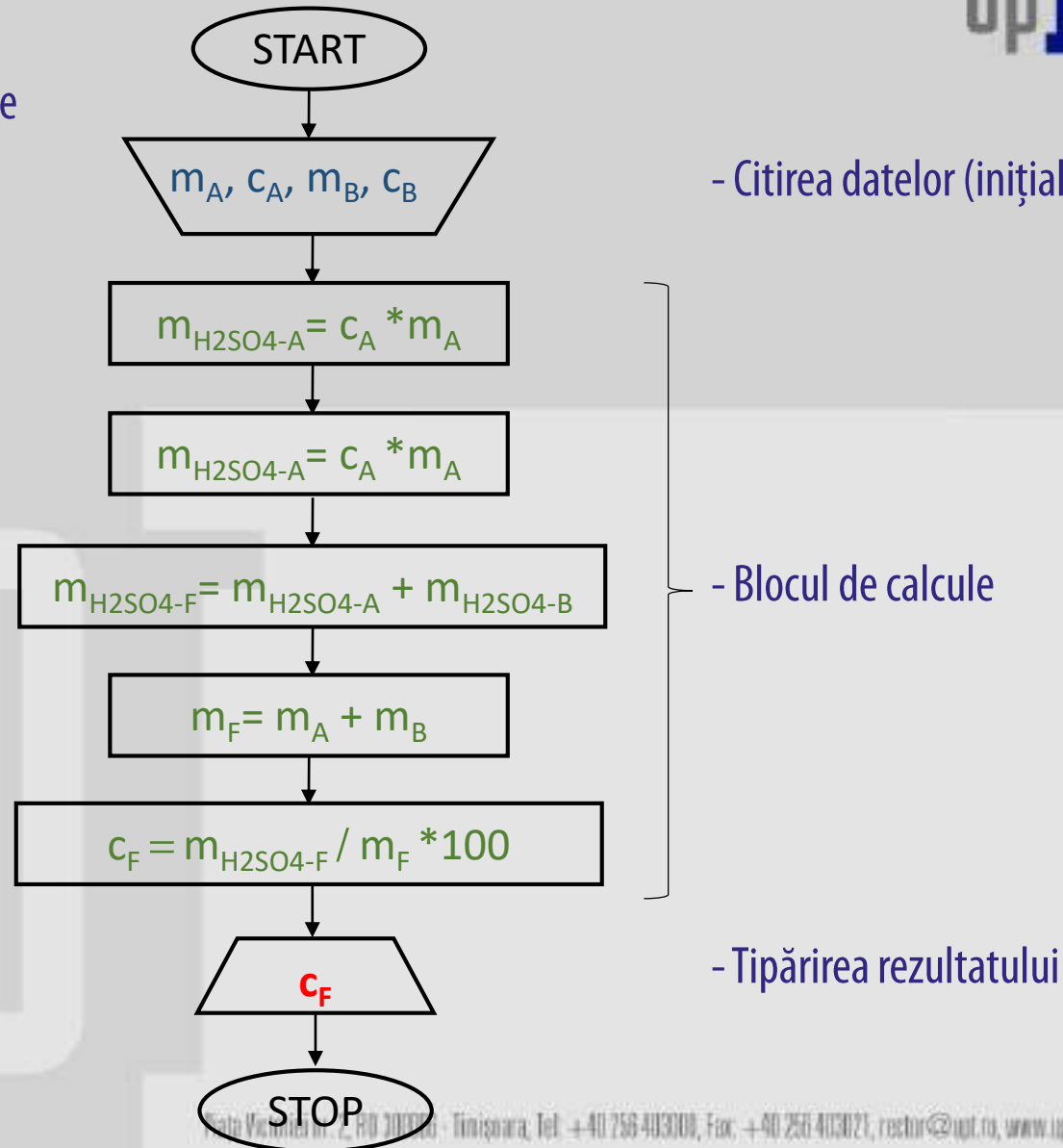
Etapele parcurse:

1. Citim primul număr
2. Citim al doilea număr
3. Calculăm suma celor două numere
4. Afișăm pe ecran suma calculată



5. Mediul de programare MATLAB

Exemplul 2. Schema logică pentru problema de calcul a concentrației:



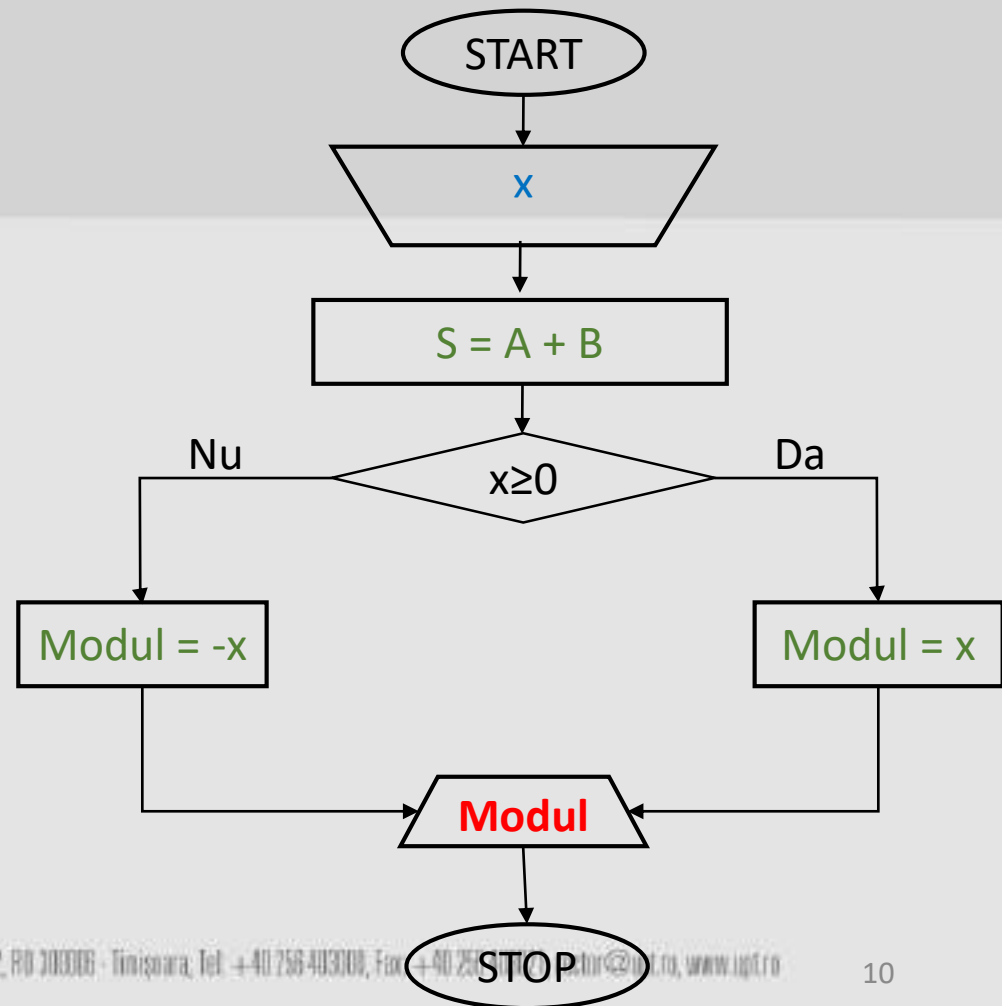
5. Mediul de programare MATLAB

Exemplul 3. Să se construiască schema logică pentru calculul modulusului unui număr real x .

$$|x| = \begin{cases} x, & x \geq 0 \\ -x, & x < 0 \end{cases}$$

Etapele parcurse:

1. Citim numărul
2. Dacă numărul este pozitiv :
 - atunci modulusul este egal cu numărul
 - altfel modulusul este egal cu numărul cu semn schimbat.
3. Afișăm pe ecran suma calculată



5. Mediul de programare MATLAB

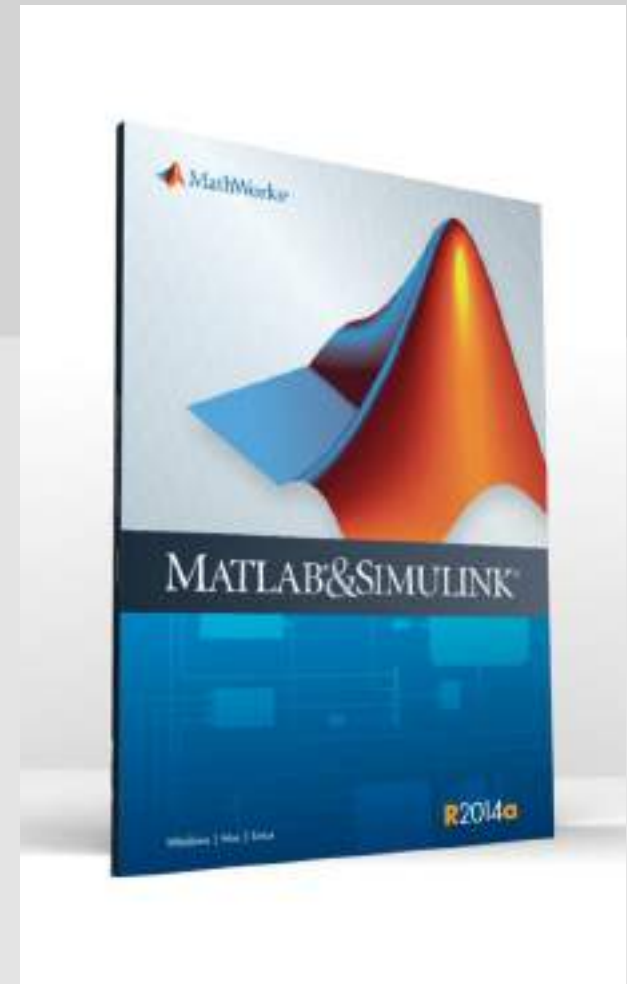
5.3. Introducere MATLAB

MATLAB® (abrevierea de la **MATrix LABoratory**) - este un limbaj de programare de înaltă performanță și în același timp un mediu interactiv pentru calcule numerice, vizualizare și programare.

MATLAB permite: analiza datelor, dezvoltarea de algoritmi, crearea de modele și aplicații.

MATLAB și-a găsit utilizarea în:

- Matematică și calcul numeric;
- Programare și dezvoltare de algoritmi;
- Modelare și simulare;
- Analiză de date, exploatare de rezultate și vizualizare;
- Dezvoltare de aplicații software, incluzând construcție de interfețe grafice cu utilizatorul (GUI) .



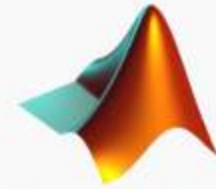
5. Mediul de programare MATLAB

MATLAB - a apărut la sfârșitul anilor ' 70.

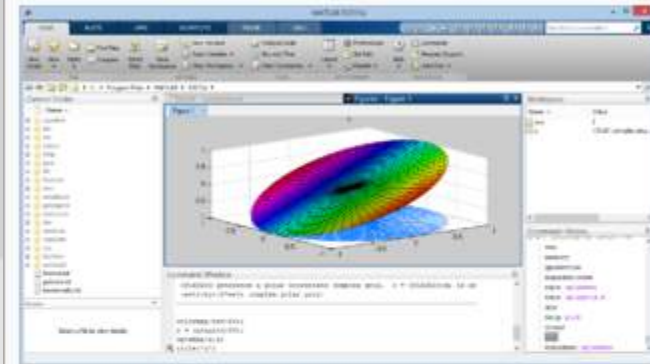
- creat de Cleve Moler, președintele departamentului de informatică al Universității din New Mexico, pentru a permite accesul studenților săi la librării specifice fără a studia limbajul FORTRAN.

- 1983 – Jack Little a început o colaborare cu Cleve Moler și Steve Bangert rescriind Matlab în limbajul C.

- 1984 – au fondat MathWorks.



L-shaped membrane logo^[1]



MATLAB R2013a running on Windows 8

Developer(s)	MathWorks
Initial release	1984; 30 years ago
Stable release	R2014b / October 3, 2014; 35 days ago
Preview release	None
Development status	Active
Written in	C, C++, Java, MATLAB
Operating system	Cross-platform: Microsoft Windows, Linux, and Mac OS X ^[2]
Platform	IA-32, x86-64
Type	Technical computing
License	Proprietary commercial software
Website	MATLAB product page

5. Mediul de programare MATLAB

Evoluția MATLAB a condus la utilizarea sa:

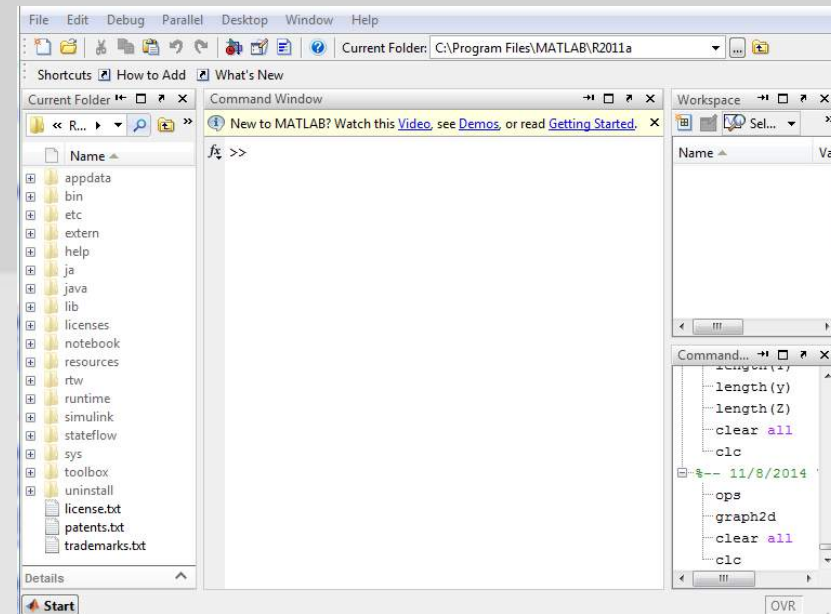
- în mediul universitar unde este pachetul standard pentru cursurile introductive și avansate de matematică, inginerie și științe;
- în industrie, unde este utilizat pentru cercetarea de înalt randament, dezvoltare și producție.

MATLAB – conține module specifice = **toolbox-uri** care permit rezolvarea unor aplicații din diverse domenii: procesarea numerică a semnalelor, sisteme de conducere automată, rețele neurale, logică fuzzy, wavelet, simulare (SIMULINK), identificare etc.

5. Mediul de programare MATLAB

Sistemul MATLAB este divizat în cinci părți principale:

1. **Limbajul MATLAB**
2. **Mediul de lucru MATLAB**
3. **Handle Graphics®**
4. **Biblioteca de funcții matematice a MATLAB-ului**
5. **Interfața de aplicații program a MATLAB-ului (API)**



5. Mediul de programare MATLAB

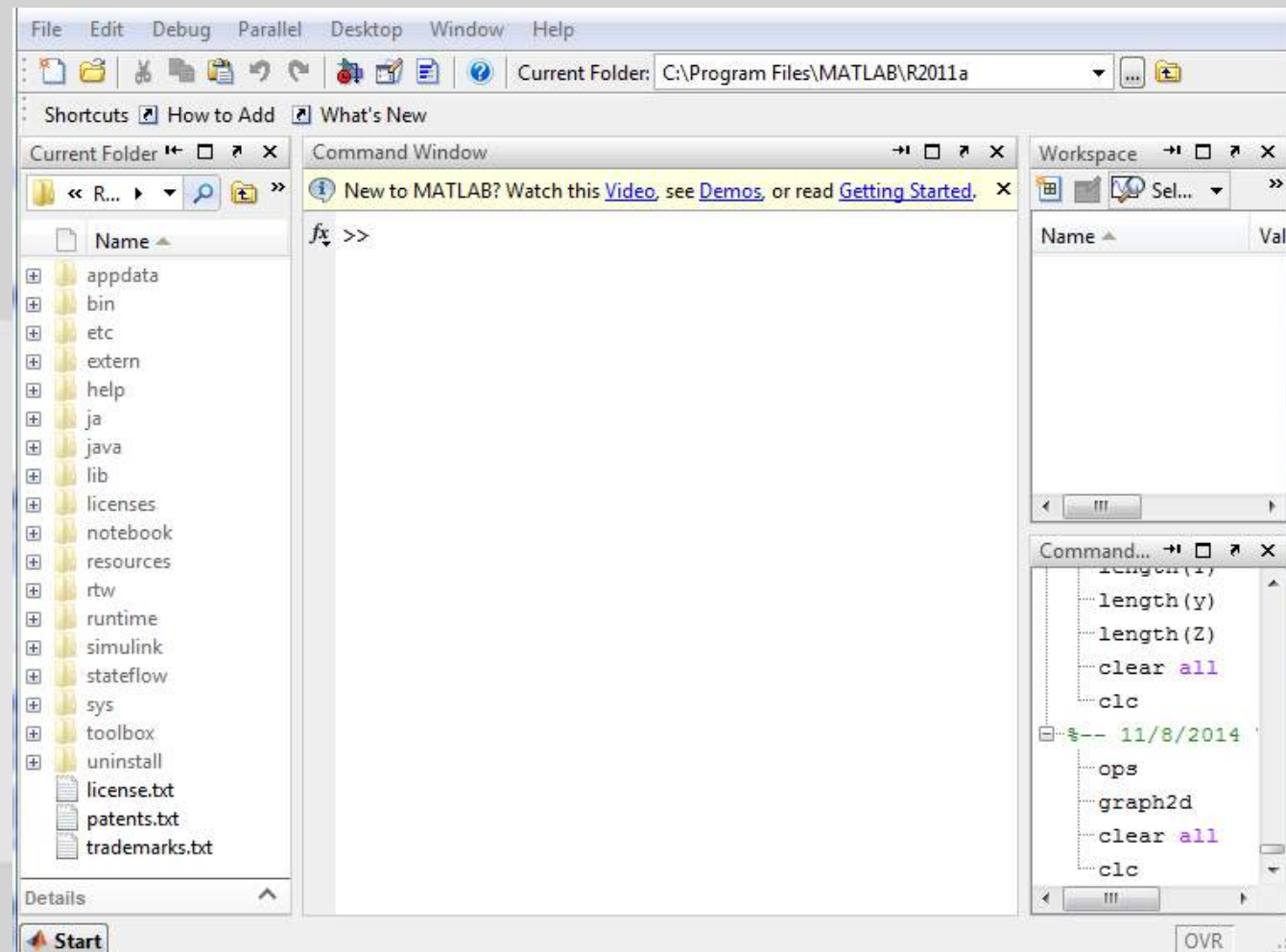
5.4. Deschidere MATLAB

Matlab-ul rulează sub Windows, iar apelarea lui se face selectând cu mouse-ul pictograma Matlab sau indicându-i calea.



Apare fereastra principală
(MATLAB Desktop)

Dacă fereastra nu se
deschide la fel, se vor realiza
următoarele operații:
Desktop → Desktop Layout
→ Default



5. Mediul de programare MATLAB

5.5. Ferestre MATLAB

1. Fereastra de comenzi
2. Directorul curent
3. Istoricul comenzilor
4. Workspace
5. Fereastra de editare
6. Fereastra grafică

The screenshot displays the MATLAB desktop environment with several windows open and numbered to match the list:

- 1. Fereastra de comenzi (Command Window):** Shows a table of numerical data with two columns.
- 2. Directorul curent (Current Folder):** Displays the file explorer for the current directory.
- 3. Istoricul comenzilor (Command History):** Lists the sequence of commands entered in the Command Window.
- 4. Workspace:** Shows the variables currently in memory, including their names and values.
- 5. Fereastra de editare (Editor):** Shows the source code for the current script.
- 6. Fereastra grafică (Figure):** Displays a 3D scatter plot of data points.

5. Mediul de programare MATLAB

1. Fereastra de comenzi

- Reprezintă principala modalitate de interacțiune între utilizator și softul MATLAB.
- Permite introducerea unor comenzi ce sunt procesate imediat, deschiderea altor ferestre, rularea de programe. Scrierea programelor în această fereastră nu este recomandată.
- Rezultatele obținute în urma rulării programelor complexe sunt afișate în fereastra de comenzi.

2. Directorul curent

- fereastra “current directory” afișează toate fișierele (de tip MATLAB sau nu) ce sunt localizate în directorul pe care MATLAB îl folosește la momentul respectiv.
- Pentru ca un program să ruleze corect, toate fișierele la care face apel trebuie să se găsească în același director.

5. Mediul de programare MATLAB

3. Istoricul comenzilor


- Înregistrează toate comenzile introduse în fereastra de comenzi.

4. Workspace

- afișează informații (dimensiune, minim, maxim, etc.) despre toate variabilele și matricele cu care se lucrează și care sunt stocate de către MATLAB.
- Permite deschiderea variabilelor și matricelor (prin dublu click pe ele) în fereastra de editare a variabilelor (Variable Editor window), care facilitează manipularea acestora.

5. Mediul de programare MATLAB

5. Fereastra de editare

- Reprezintă locul unde se scriu fișierele text (conțin linii program cu cod în limbajul MATLAB), care apoi sunt compilate și rulate în fereastra de comenzi. Aceste fișiere sunt denumite M-files pentru că **extensia** lor este **.m**.
- Nu se deschide în momentul accesării MATLAB.
- Pentru deschiderea unei ferestre noi de acest tip există mai multe posibilități:
 - File → New → Script
 - Apăsarea combinației de taste: Ctrl + N
 - Apăsarea icoanei  din partea stângă sus a ecranului

5. Mediul de programare MATLAB

5. Fereastra de editare

Fișierele se salvează astfel: File → Save as ... → Nume.

OBS. Numele fișierului trebuie să aibă ca prim caracter o literă, urmată de litere, cifre sau “_”.

Rularea fișierului salvat se poate face în două moduri:

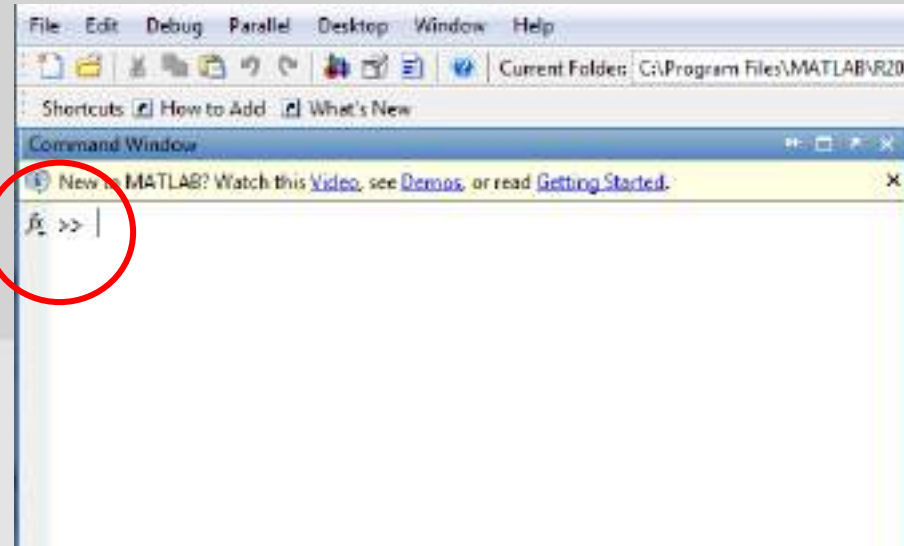
- din fereastra de editare: Debug → Run;
- din fereastra de comenzi: prin tastarea numelui fișierului în dreptul cursorului urmat de apăsarea tastei Enter.

6. Fereastra grafică – la utilizarea instrumentelor Matlab de reprezentare grafică, rezultatul este redat într-o fereastră separată, numită fereastra grafică.

5. Mediul de programare MATLAB

5.6. Manipularea ferestrei de comenzi

- La deschiderea MATLAB se afișează simbolul `>>`;
- Pentru a scrie o comandă cursorul trebuie să fie amplasat lângă acest simbol;
- După scrierea comenzii se apasă tasta **Enter** și comanda este executată și afișată.
- Se pot scrie mai multe comenzi pe aceeași linie. Separarea între acestea se face prin virgulă ",". La apăsarea tastei Enter comenzile sunt executate în ordine de la stânga la dreapta.
- O comandă scrisă anterior poate fi reluată în fereastra de comenzi prin utilizarea tastei "↑". Odată apărută în fereastră, comanda poate fi modificată și executată.
- Dacă o comandă este atât de lungă încât nu intră într-o linie, ea poate fi continuată pe linia următoare, astfel: se scriu 3 puncte pe linia curentă "...", se apasă tasta **Enter** și se continuă comanda pe noua linie.



5. Mediul de programare MATLAB

- **Punctul și virgula (";")**: dacă la finele unei comenzi scrise în fereastra de comenzi se pune ";" și se apasă Enter ⇒ comanda este procesată, dar **rezultatul nu mai este afișat**. Acest lucru este util atunci când avem o succesiune de comenzi și ne interesează doar rezultatul final.
 - dacă mai multe comenzi sunt scrise în aceeași linie, urmate de ";", rezultatele nu vor mai fi afișate.
- **Simbolul "%"** : când simbolul procent este plasat la începutul unui mesaj ⇒ mesajul reprezintă un comentariu. De obicei comentariile se fac în cadrul fișierelor realizate în fereastra de editare.
- Comanda **"clc"**: șterge toate informațiile din fereastra de comenzi, dar nu acționează asupra variabilelor memorate de program.
- Comanda **"clear all"**: șterge toate variabilele din memoria MATLAB (curăță workspace).

5. Mediul de programare MATLAB

5.7. Definirea variabilelor

Variabila – reprezintă un nume căruia îi este atribuită o valoare numerică. După ce variabilei îi este atribuită o valoare numerică, acesta poate fi utilizată în orice linie de comandă Matlab. La definirea unei variabile, Matlabul alocă un spațiu de memorie unde valoarea acesteia este stocată. Dacă variabilei îi este atribuită o altă valoare, conținutul din memoria Matlab este înlocuit.

Operatorul ce realizează atribuirea unei valori numerice la o variabilă este semnul egal "=".

Sintaxa de atribuire este: **nume_variabila = valoare numerică sau expresie**

```
>> a=6
```

```
a =
```

```
6
```

```
>> b=3*a+24
```

```
b =
```

```
42
```

5. Mediul de programare MATLAB

Reguli de denumire a variabilelor

- Numele variabilei trebuie să înceapă cu o literă;
- Numele variabilei poate conține: litere, cifre și liniuța de subliniere " _ " (underscore);
- Lungimea numelui unei variabile poate fi de maxim 63 caractere și variază în funcție de versiunea de MATLAB.
- MATLAB-ul face diferența între litere mari și litere mici. Prin urmare: ab, Ab, aB și AB reprezintă 4 variabile distincte.
- Nu se vor denumi variabilele cu nume de funcții deja definite în MATLAB (sin, exp, etc.). Odată ce numele este atribuit variabilei, funcția nu mai poate fi utilizată.

OBS. Aceste reguli se păstrează și la denumirea fișierelor MATLAB.