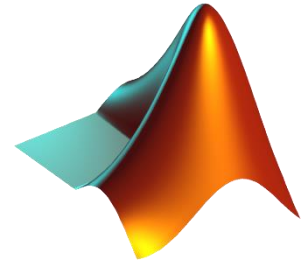


Εισαγωγή στο MATLAB

Γιώργος Γιαννακάκης

Το **MATLAB** (MATrix LABoratory) αποτελεί ένα εύχρηστο, ανοιχτού κώδικά υπολογιστικό περιβάλλον για υλοποίηση επιστημονικών εφαρμογών σε ένα φάσμα πεδίων, όπως Γραμμική Άλγεβρα, Μαθηματική Ανάλυση, Αριθμητική Ανάλυση, Θεωρία Ελέγχου και Επεξεργασία Σημάτων και Εικόνας. Το μεγάλο πλήθος ενσωματωμένων συναρτήσεων (built in functions) καθώς και εξωτερικών βιβλιοθηκών (toolboxes) το καθιστούν ένα ισχυρό εργαλείο για πολλές και εξειδικευμένες περιοχές εφαρμογών. Τα αρχεία που χρησιμοποιούνται από το MATLAB είναι αρχεία δεδομένων (.mat, .txt, .dat κλπ), αρχεία γραφημάτων (.fig), αρχεία δέσμης εντολών (.m files) καθώς και αρχεία συναρτήσεων (.m functions). Αποτελεί μια απλή και δομημένη γλώσσα προγραμματισμού (script language) χωρίς αυστηρό προγραμματιστικό συντακτικό για χρήση και διασύνδεση των παραπάνω.



Μεταβλητές

Οι μεταβλητές ορίζονται με τον ακόλουθο τρόπο

```
>> a=5; A=zeros(3,4);
```

Προσοχή: υπάρχει διάκριση μεταξύ μικρών και κεφαλαίων γραμμάτων.

Καλή πρακτική είναι η δήλωση μεταβλητών πριν την χρησιμοποίησή τους για αποδοτική διαχείριση μνήμης.

Αριθμητικές Πράξεις

Οι βασικές αριθμητικές πράξεις στο περιβάλλον MATLAB:

- + πρόσθεση
- αφαίρεση
- * πολλαπλασιασμός
- .* πολλαπλασιασμός στοιχείο προς στοιχείο
- / διαίρεση
- ./ διαίρεση στοιχείο προς στοιχείο
- ^ ύψωση σε δύναμη, π.χ. $x^2 = x^2$

π.χ

```
>> a*b
>> c/5
>> d^2
>> [1 2 3]*[1 2 3]'
ans =
    14
>> [1 2 3].*[1 2 3]
ans =
     1     4     9
```

Πίνακες

Η δημιουργία πινάκων στο MATLAB έχει ως εξής

```
>> A=zeros(3,4); %μηδενικός πίνακας 3x4
>> A=ones(2,5); %μονικός πίνακας 2x5
>> A=eye(4); %μοναδιαίος πίνακας 4x4
>> A=magic(5); %μαγικός πίνακας 5x5
>> A=[1 2; 3 4]
A =
1 2
3 4
>> x=[1:10]
x = 1 2 3 4 5 6 7 8 9 10
```

Οι θέσεις των στοιχείων πίνακα προσπελάζονται

```
>> x(3) %το τρίτο στοιχείο του x
>> A(2,3) %το στοιχείο της 2ης γραμμής και της τρίτης στήλης του πίνακα A
>> A(2,:) %η δεύτερη γραμμή του πίνακα A
>> A(:,1) %η πρώτη στήλη του πίνακα A
>> A(1:3,4) %τα τρία πρώτα στοιχεία της 4ης στήλης
```

Πράξεις σε πίνακες

Συνάρτηση	Διαδικασία
size(A)	Διαστάσεις του πίνακα A
det(A)	Ορίζουσα πίνακα A
max(A)	Εύρεση μέγιστου κάθε στήλης του πίνακα A
min(A)	Εύρεση ελάχιστου κάθε στήλης του πίνακα A
sum(A)	Άθροισμα κάθε στήλης του πίνακα A
mean(A)	Μέσος όρος κάθε στήλης του πίνακα A
diag(A)	Στοιχεία κύριας διαγωνίου πίνακα A
A'	Ανάστροφος πίνακας A^T
inv(A)	Αντίστροφος πίνακας A^{-1}

Λογικοί/Σχετικοί Τελεστές

Οι λογικοί τελεστές στο MATLAB είναι

- > : μεγαλύτερο από
- < : μικρότερο από
- >=: μεγαλύτερο ίσο από
- <=: μικρότερο ίσο από
- ==: είναι ίσο με
- ~= : δεν είναι ίσο με
- | : λογικό OR
- & : λογικό AND
- ~ : λογικό NOT

π.χ

```
>> R = S1 & S2
>> R = ~S
>> if (a>b) & (c<d)
```

Συνάρτηση	Διαδικασία
max	μέγιστο στοιχείο διανύσματος
min	ελάχιστο στοιχείο διανύσματος
length	μήκος διανύσματος
sort	ταξινόμηση σε αύξουσα σειρά
sum	άθροισμα στοιχείων
prod	γινόμενο στοιχείων
norm	νόρμα διανύσματος
median	διάμεσος
mean	μέση τιμή
std	τυπική απόκλιση

Ελεγχος ροής

Ροή με την εντολή if

Ακολουθεί τη σύνταξη

```

if συνθήκη_ελέγχου
    εντολή ;
:
    εντολή ;
end

```

Αν ισχύει η συνθήκη ελέγχου εκτελείται ο κώδικας που είναι μεταξύ if και end

Ροή με την εντολή if ... elseif ... else

Ακολουθεί τη σύνταξη

```

if συνθήκη1
    μπλοκ εντολών
elseif συνθήκη2
    μπλοκ εντολών2
elseif συνθήκη3
    μπλοκ εντολών3
:
else
    μπλοκ εντολών
end

```

Για την κατασκευή σύνθετων δομών if χρησιμοποιείται η εντολή elseif, η οποία δεν απαιτεί στο τέλος της end. Ο έλεγχος γίνεται μόνο εφόσον αποτυγχάνει η συνθήκη του if και οποιουδήποτε προγενέστερου elseif.

Δημιουργία βρόχων

Βρόχοι for

```

for i=1:10,
    for j=1:10,
        t(i,j) = i/j;
    end
end
end

```

Βρόχοι while

```
while(i<max(size(x)))
    y(i+1) = y(i) + h*(x(i)-abs(y(i)));
    i = i+1;
end
```

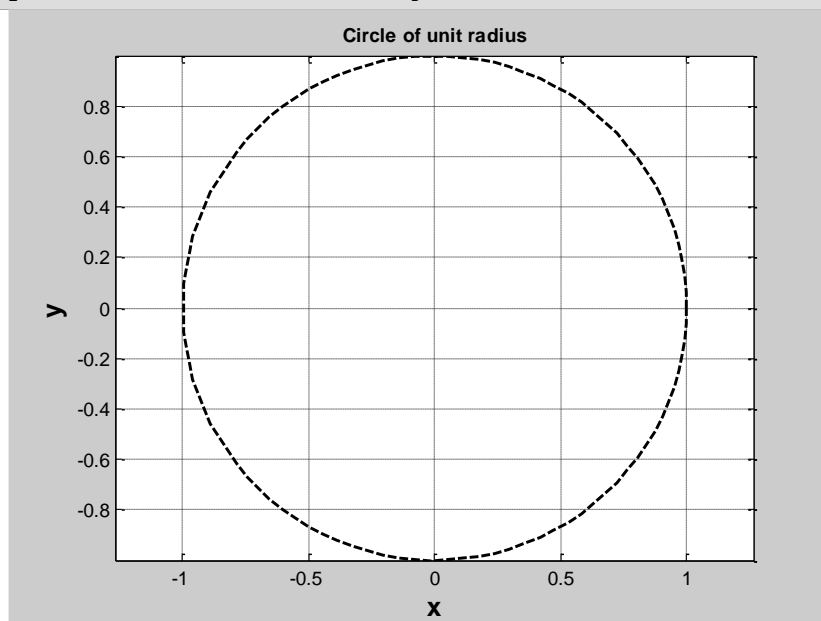
Με την εντολή `break` τερματίζεται και εξάγεται η ροή του προγράμματος από βρόχο.

```
>> for(i=1:10)
>> x(i)=2*i+3*sin(2*pi*5*i/10);
>> end
>> i=1;
>> while(i<7)
>> a(i)=5*(i-1);
>> i=i+1;
>> end
>> if(a>5 & b<1)
>> C=3;
>> elseif(a<2 & b>4)
>> C=4;
>> else
>> C=0;
>> end
```

Γραφήματα

```
>> a=linspace(0,2*pi,100);
>> x=cos(a);
>> y=sin(a);

>> figure;
>> plot(x,y,'k--','LineWidth',2); %Δημιουργία γραφήματος
>> axis('equal'); grid on; %Τροποποίηση αξόνων
>> title('Circle of unit radius','Fontweight','bold'); %Τίτλος
>> xlabel('x','FontSize',14,'FontWeight','demi'); %Τίτλος x άξονα
>> ylabel('y','FontSize',14,'FontWeight','demi'); %Τίτλος y άξονα
```



Υλοποίηση συνάρτησης

Η χρήση δομημένου προγραμματισμού μέσω συναρτήσεων αποτελεί μία καλή προγραμματιστική τεχνική ιδίως όταν ο κώδικας επαναχρησιμοποιείται. Το MATLAB δίνει την δυνατότητα δημιουργίας συναρτήσεων οι οποίες δηλώνονται

```
function z = my_func(x, y)
```

όπου my_func είναι το όνομα της συνάρτησης, x,y είναι τα ορίσματα εισόδου και z η μεταβλητή εξόδου.

Να δημιουργήσετε την συνάρτηση

$$x(n) = 5 \cdot \sin(2\pi 10n) + e^{-2n} \quad n = -2, -1.99, \dots, 1.99, 2$$

καθώς και το διάγραμμα του

Παρατίθεται ο κώδικας MATLAB που πρέπει να συμπληρώσετε

```
n=...
x=zeros(...,1);
for(i=1:...)
    x(i,1)=...
end
figure; plot(...);
```

MATLAB και Εικόνες

Η εικόνα στο MATLAB αντιμετωπίζεται ως δισδιάστατος πίνακας

Τύποι Εικόνας :

Τύπος	Περιγραφή	Εύρος τιμών
int8	8-bit integer	-128 ÷ 127
uint8	8-bit unsigned integer	0 ÷ 255
int16	16-bit integer	-32768 ÷ 32767
uint16	16-bit unsigned integer	0 ÷ 65535
double	Double precision real number	Machine specific

Εντολές διαχείρισης Εικόνων

Ανάγνωση Εικόνων

```
>> C=imread(image)
```

ανάγνωση εικόνας image π.χ. C=imread('name.tif').

Μετατροπή ενός πίνακα σε εικόνα

```
>> imwrite(C,'name',fmt)
```

όπου fmt είναι το επιθυμητό format της εικόνας

Δημιουργία παραθύρου γραφήματος

```
>> figure(1);
```

δημιουργία νέου παραθύρου γραφικών (χωρίς όρισμα ή με όρισμα έναν ακέραιο).

Απεικόνιση εικόνας στην οθόνη

```
>> imshow(image, [low high])
```

εμφάνιση εικόνας στο ενεργό παράθυρο γραφικών (ή σε νέο, αν δεν υπάρχει). Το πρώτο όρισμα είναι ο πίνακας που περιέχει τις τιμές φωτεινότητας. Το δεύτερο, αν υπάρχει, είναι ένα διάνυσμα δύο τιμών, όπου η πρώτη είναι η ελάχιστη τιμή φωτεινότητας (που απεικονίζεται ως μαύρο) και η δεύτερη είναι η μέγιστη (που απεικονίζεται ως λευκό).

Πληροφορίες εικόνας

```
>> imfinfo('cameraman.tif')
ans =

        Filename: 'C:\Program
Files\MATLAB\R2012a\toolbox\images\imdemos\cameraman.tif'
      FileModDate: '04-Dec-2000 14:57:54'
        FileSize: 65240
         Format: 'tif'
   FormatVersion: []
           Width: 256
           Height: 256
        BitDepth: 8
        ColorType: 'grayscale'
FormatSignature: [77 77 0 42]
      ByteOrder: 'big-endian'
NewSubFileType: 0
  BitsPerSample: 8
    Compression: 'PackBits'
PhotometricInterpretation: 'BlackIsZero'
   StripOffsets: [8x1 double]
SamplesPerPixel: 1
   RowsPerStrip: 32
StripByteCounts: [8x1 double]
  XResolution: 72
  YResolution: 72
ResolutionUnit: 'None'
        Colormap: []
PlanarConfiguration: 'Chunky'
      TileWidth: []
      TileLength: []
      TileOffsets: []
      TileByteCounts: []
      Orientation: 1
        FillOrder: 1
GrayResponseUnit: 0.0100
  MaxSampleValue: 255
  MinSampleValue: 0
      Thresholding: 1
           Offset: 64872
ImageDescription: [1x112 char]
```

παρουσιάζει πληροφορίες για την εικόνα

Ιστόγραμμα εικόνας

```
>> imhist(A, n)
```

υπολογισμός ιστογράμματος εικόνας A όπου n είναι ο αριθμός των στηλών του ιστογράμματος.

Εκτύπωση εικόνας

```
>> print [-ddevice] [-options] <filename>
```

Η εκτύπωση πραγματοποιείται με παραμέτρους την συσκευή εξόδου, άλλες παραμέτρους εκτύπωσης, όνομα αρχείου.

Βασικές λειτουργίες εικόνων

Συνάρτηση	Διαδικασία
<code>imabsdiff(I, J)</code>	Αφαιρεί την εικόνα J από την εικόνα εικονοστοιχείο προς εικονοστοιχείο και επιστρέφει την διαφορά τους
<code>imcomplement(I)</code>	Συμπληρωματικό της εικόνας I
<code>imadd(I, J)</code>	Προσθέτει τις εικόνες I και J
<code>imsubtract(I, J)</code>	Αφαιρεί την εικόνα J από την I
<code>imdivide(I, J)</code>	Διαιρεί την εικόνα I σε J
<code>immultiply(I, J)</code>	Πολλαπλασιάζει την εικόνα I και J

Μετατροπή έγχρωμης εικόνας σε γκριζου τόνου

```
A=imread('autumn.tif');  
B=rgb2gray(A);  
figure;  
subplot(1,2,1); imshow(A); subplot(1,2,2); imshow(B);
```



Εξαγωγή καναλιών χρώματος

```
>> A=imread('autumn.tif');  
>> Ared=A(:,:,1); %extract red channel  
>> Agreen=A(:,:,2); %extract green channel  
>> Ablue=A(:,:,3); %extract blue channel  
>> subplot(2,2,1); imshow(A); axis image;  
>> subplot(2,2,2); imshow(Ared); title('red');  
>> subplot(2,2,3); imshow(Agreen); title('green');  
>> subplot(2,2,4); imshow(Ablue); title('blue');
```

