

Μέθοδοι Επίλυσης με Η/Υ

9^ο Μάθημα: *Γραφή/ανάγνωση μεταβλητών
σε αρχεία*

Ν.Δ. Λαγαρός, Α. Στάμος, Χ. Φραγκουδάκης

Ανάγνωση και γραφή σε αρχεία

Το Matlab διαβάζει και αποθηκεύει από/σε αρχεία δεδομένων διαφόρων μορφών

- αρχεία με κατάληξη `.mat`
διαβάζονται μόνο από το Matlab
`save('ΌνομαΑρχειου.mat','var_1','var_2','var_3')`
- απλά αρχεία κειμένου (αρχεία ASCII)
διαβάζονται από όλους τους editors κειμένου (Σημειωματάριο, Word, κλπ) εντολές: `dlmread` και `dlmwrite`
- αρχεία Excel ή/και csv (comma separated)
(εντολές `xlsread` και `xlswrite`)
- μορφοποιημένα αρχεία
Ο ισχυρότερος τρόπος γραφής και ανάγνωσης δεδομένων από αρχεία ASCII (εντολές: `fprintf`, `sprintf`)

σημερινό μάθημα

Η εντολή sprintf

`keim = sprintf(διαμόρφωση, A1, ..., An)`

π.χ. `sprintf('%5.3f',pi)` δίνει αποτέλεσμα `3.142`

όπου η «διαμόρφωση» έχει τη μορφή:

% πλατοςΣυνολικό.δεκαδικάΨηφία f 5 ψηφία συνολικά και 3 δεκαδικά

e: επιστημονική γραφή, **% πλατοςΣυνολικό.σημαντικάΨηφία e**

αντί για f (float number) μπορούμε να χρησιμοποιήσουμε το “e” (σημαίνει επιστημονική γραφή, χρήσιμο για να γράφουμε μεγάλους αριθμούς)

π.χ. `sprintf('%5.3e',1000*pi)` δίνει `3.142e+03` (5 ψηφία, 3 δεκαδικά)

g: συμπαγής γραφή, **% πλατοςΣυνολικό.σημαντικάΨηφία g**

π.χ. δοκιμάστε `sprintf('%g',pi)`,

ενώ, `sprintf('%5.3g',pi)` δίνει `3.14` (3 σημαντικά ψηφία)

Γραφή σε μορφοποιημένα αρχεία

`keim = sprintf(διαμόρφωση, A1, ..., An)`

Παράδειγμα:

```
a0 = 1/exp(1);           % A = 1/e (βάση του φυσικού λογαρίθμου)
```

```
str_e = sprintf('to a0 einai iso me %0.5e', a0)
```

```
str_f = sprintf('to a0 einai iso me %0.5f', a0)
```

```
str_g = sprintf('to a0 einai iso me %0.5g', a0)
```

```
str_e =
```

```
To a0 einai iso me 3.67879e-01
```

```
str_f =
```

```
To a0 einai iso me 0.36788
```

```
str_g =
```

```
To a0 einai iso me 0.36788
```

Γραφή σε μορφοποιημένα αρχεία

`fprintf(αρχείο, διαμόρφωση, A1, ... ,An)`

η εντολή **fprintf** διαφέρει από την **sprintf** στο ότι το αποτέλεσμα γράφεται σε ένα αρχείο και όχι σε μία μεταβλητή

```
A = [6,12,4;7.567,-2,3;2,8.55,-9.2]
```

```
fid =fopen('mitrwoA.txt','w'); % ανοιγούμε το αρχείο στο οποίο θα γράψουμε  
fprintf(fid,' Ektypwsi toy mitrwou A: \n'); % κάποιο κείμενο  
fprintf(fid,'-----\n'); % κάποιο κείμενο
```

% το \n λέει στο πρόγραμμα να αλλάξει γραμμή

```
for i=1:3  
    % αρχείο |<----- διαμόρφωση ----->| <----- μεταβλητές ----->|  
    fprintf(fid,'[%10.4f %10.4f %10.4f] \n',A(i,1),A(i,2),A(i,3));  
end  
fclose(fid) % κλείνουμε το αρχείο – ώστε να μπορεί να ανοίξει από άλλα προγράμματα
```

Το αρχείο mitrwoA.txt θα είναι:

```
Ektypwsi toy mitrwou A:
```

```
-----  
[      6.0000      12.0000       4.0000]  
[      7.5670      -2.0000       3.0000]  
[      2.0000       8.5500      -9.2000]
```

Η εντολή fprintf

Παράδειγμα: `fprintf(αρχείο, διαμόρφωση, A1, ... ,An)`

Το προηγούμενο πρόγραμμα θα μπορούσε να γραφεί με την παρακάτω μορφή (αποφεύγοντας την δομή επανάληψης for):

```
A = [6,12,4;7.567,-2,3;2,8.55,-9.2]
fid =fopen('mitrwoA.txt','w');
fprintf(fid,' Ektypwsi toy mitrwou A: \n');
fprintf(fid,'-----\n');
fprintf(fid,'[%10.4f %10.4f %10.4f] \n',A(1,:),A(2,:),A(3,:));
fclose(fid)
```

ή εκμεταλλευόμενοι ότι τα μητρώα τυπώνονται κατά στήλες:

```
A = [6,12,4;7.567,-2,3;2,8.55,-9.2]
fid =fopen('mitrwoA.txt','w');
fprintf(fid,' Ektypwsi toy mitrwou A: \n');
fprintf(fid,'-----\n');
fprintf(fid,'[%10.4f %10.4f %10.4f] \n',A');
fclose(fid)
```

Γραφή σε μορφοποιημένα αρχεία

Παράδειγμα:

Έστω ότι το διάνυσμα/μεταβλητή `Acc` περιέχει τιμές επιτάχυνσης με χρονικό βήμα `0.02sec`. Να γραφεί η μεταβλητή `Acc` σε ένα αρχείο ASCII με διαμόρφωση:

α. στην 1^η γραμμή να δίνονται το πλήθος τιμών (ακέραιος αριθμος) και το χρονικό βήμα (πραγματικός αριθμός με τρία δεκαδικά ψηφία)

β. στις υπόλοιπες γραμμές να δίνονται οι τιμές της επιτάχυνσης, έχοντας 5 τιμές σε κάθε γραμμή.

% ανάγνωση του αρχείου

```
Acc=dlmread('earthquake1.dat');
```

% γραφή στο αρχείο seismos.txt με νέα διαμόρφωση της επιλογής μας

```
dt=0.02;
```

```
nsteps = length(Acc);
```

```
fid =fopen('seismos.txt','w');
```

```
fprintf(fid,'time step: %g, number of steps: %g \n',dt,nsteps);
```

```
fprintf(fid,'%14.6e, %14.6e, %14.6e, %14.6e, %14.6e \n',Acc(:));
```

```
fclose(fid)
```

αρχείο `seismos.txt` θα είναι:

```
time step: 0.02, number of steps: 3000
```

```
 6.631000e+00,  3.507000e+00,  1.628000e+00,  1.105000e+00,  4.260000e-01  
 2.390000e-01, -2.680000e-01, -1.836000e+00, -1.490000e+00,  1.413000e+00  
-1.058000e+00,  1.669000e+00,  1.167600e+01,  1.149800e+01, -2.920000e-01  
-1.671800e+01, -1.228200e+01, -3.280000e+00, -1.202900e+01, -6.913000e+00
```

Ανάγνωση αρχείων τυχαίας διαμόρφωσης

Έστω το αρχείο `seismos.txt` με περιεχόμενο:

```
time step: 0.02, number of steps: 3000
 6.631000e+00,  3.507000e+00,  1.628000e+00,  1.105000e+00,  4.260000e-01
 2.390000e-01, -2.680000e-01, -1.836000e+00, -1.490000e+00,  1.413000e+00
-1.058000e+00,  1.669000e+00,  1.167600e+01,  1.149800e+01, -2.920000e-01
-1.671800e+01, -1.228200e+01, -3.280000e+00, -1.202900e+01, -6.913000e+00
```

Θέλουμε:

(α) να διαβάσουμε όλες τις τιμές του αρχείου και να τις αποθηκεύσουμε στο διάνυσμα `acc`

(β) να διαβάσουμε το βήμα και το μήκος της καταγραφής και να τα αποθηκεύσουμε στις μεταβλητές `dt` και `nsteps`

Ανάγνωση αρχείων τυχαίας διαμόρφωσης

Το αρχείο seismos.txt διαβάζεται με τον παρακάτω κώδικα:

```
Clear; clc; close all;
fid =fopen('seismos.txt');
tline = fgetl(fid);

tmp = sscanf(tline,'time step: %f, number of steps: %f');
dt = tmp(1);
nsteps = tmp(2);

count=0;
while feof(fid) == 0
    tline = fgetl(fid);
    tmp = sscanf(tline,'%f, %f, %f, %f, %f')
    acc(count+1:count+5) = tmp;
    count = count + 5;
end
fclose(fid)

plot((1:length(acc))*dt,acc,'b.-')
```

Ανάγνωση αρχείων τυχαίας διαμόρφωσης

αναλυτικά ο κώδικας:

ανοίγουμε το αρχείο που θα διαβάσουμε, το Matlab εφεξής θα το αναγνωρίζει ως fid

```
fid = fopen('seismos.txt');
```

εντολή fgetl διαβάζει ως μία συμβολοσειρά την πρώτη γραμμή του αρχείου

```
tline = fgetl(fid);
```

με την εντολή sscanf διαβάζουμε τα περιεχόμενα της 1^{ης} γραμμής

```
tmp = sscanf(tline, 'time step: %f, number of steps: %f');
```

```
dt = tmp(1);
```

```
nsteps = tmp(2);
```

Ανάγνωση αρχείων τυχαίας διαμόρφωσης

```
count=0;
```

Λέμε στο πρόγραμμα να κάνει επαναλήψεις μέχρι να βρεί το τέλος του αρχείου fid (εντολή `feof(fid) == 0`)

```
while feof(fid) == 0
```

διαβάζουμε κάθε γραμμή μία-μία (αφού βρισκόμαστε μέσα σε ένα for loop)

```
tline = fgetl(fid);
```

εξάγουμε τα αριθμητικά δεδομένα και τα αποθηκεύουμε στο 5x1 διάνυσμα tmp

```
tmp = sscanf(tline, '%f, %f, %f, %f, %f');
```

αποθηκεύουμε τα αριθμητικά δεδομένα στο διάνυσμα acc

```
acc(count+1:count+5) = tmp;
```

προσαυξάνουμε τον μετρητή count κατά 5 (τόσες τιμές διαβάσαμε)

```
count = count + 5;
```

```
end
```

Δεν ξεχνάμε στο τέλος να «κλείσουμε» το αρχείο fid (ώστε να μπορεί να το διαβάσει κάποιο άλλο πρόγραμμα).

```
fclose(fid)
```

Ερωτήσεις...