

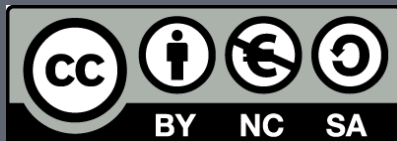


ΑΝΟΙΚΤΑ ακαδημαϊκά
μαθήματα ΠΠ



Εισαγωγή στον Προγραμματισμό Η/Υ για Χημικούς Μηχανικούς

Παρουσίαση Διαλέξεων: 5. Επικοινωνία
Καθηγητής Δημήτρης Ματαράς



Copyright © 2014 by Prof. D. S. Mataras (mataras@upatras.gr). This work is made available under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 license, <http://creativecommons.org/licenses/by-nc-nd/3.0/>

Οι βασικές προτάσεις εισόδου/εξόδου

READ, **WRITE**, **PRINT**

- ▶ *Πλήρης μορφή*, που χρησιμοποιείται γενικώς:

READ (λίστα ελέγχου) [λίστα δεδομένων]

WRITE (λίστα ελέγχου) [λίστα δεδομένων]

- ▶ *Απλουστευμένη μορφή*, που χρησιμοποιείται για την επικοινωνία με τις προκαθορισμένες μονάδες εισόδου/εξόδου:

READ μορφή[, λίστα δεδομένων]

PRINT μορφή[, λίστα δεδομένων]

λίστα δεδομένων: λίστα ονομάτων, ή ρητών σταθερών ή εκφράσεων, ή υπονοούμενων επαναλήψεων

λίστα ελέγχου: μπορεί να περιλαμβάνει έναν ή περισσότερους από τους ακόλουθους όρους:

Όρος λίστας ελέγχου

σημασία/χρήση

[UNIT=] μονάδα	ο αριθμός της λογικής μονάδας
[FMT=] μορφή	καθορισμός μορφής , μπορεί να είναι μία έκφραση ελέγχου μορφής (σε αποστρόφους ' '), αστερίσκος * (ελεύθερη μορφή), ή η ετικέτα της εντολής μορφοποίησης FORMAT
[NML=όνομα]	το όνομα μιας λίστας NAMELIST
REC=πεδίο	όπου πεδίο = ακέραιος ή έκφραση με ακέραιο αποτέλεσμα, που δείχνει την σειρά του πεδίου στο αρχείο άμεσης πρόσβασης. Η χρήση του είναι υποχρεωτική αν πρόκειται για αρχείο άμεσης πρόσβασης
[ADVANCE=]	'YES' , 'NO' , καθορίζει αν η μεταφορά δεδομένων προχωράει στο επόμενο πεδίο μετά από κάθε τιμή που διαβάζεται ή γράφεται.
[END=ετικέτα]	μόλις συναντήσει το πεδίο EOF μεταφέρει τον έλεγχο στην ετικέτα .
[EOR=ετικέτα]	μόλις συναντήσει το τέλος του τρέχοντος πεδίου μεταφέρει τον έλεγχο στην ετικέτα .
[ERR=ετικέτα]	μόλις συναντήσει σφάλμα μεταφέρει τον έλεγχο στην ετικέτα .
[IOSTAT=όνομα]	το όνομα ακέραιας μεταβλητής της οποίας η τιμή δείχνει αν έχει προκύψει σφάλμα
[SIZE=όνομα]	ακέραια μεταβλητή = το πλήθος χαρακτήρων που διαβάζει το READ με ADVANCE='NO' , πριν συναντήσει το τέλος του πεδίου.

Απλές προτάσεις με ελεύθερη μορφή

```
integer::a = 2,b = 3,c = 4
print*,a, b, c, a + 2, b * c
print*, 'give x'      ! τυπώνει: give x
read*, x              !και περιμένει την τιμή από τον χρήστη
print*, x
print*, 'give a,b,c'
read*, a, b, c      ; print*, a, b, c
!διάβασε και τύπωσε με ελεύθερη μορφή τις μεταβλητές α,β,γ
!(εννοείται από το πληκτρολόγιο και στην οθόνη αντίστοιχα)

print*, 'give a,b,c'; read(*,*)a, b, c
write(*,*)a, b, c !γράφει με ελεύθερη μορφή στην οθόνη
write(*,*) 'a*b=', a * b
end
```

Η προπαίδεια

```
integer::a, i, j
```

```
do i = 2, 10
```

Μη επαναλήψιμος μορφοποιητής

```
do j = 1, 10
```

επαναλήψιμος μορφοποιητής

```
  a = j * i
```

```
  print '($,i5)', a
```

```
  write (*, '($,i5)') a ! το ίδιο με το προηγούμενο
```

```
enddo
```

```
print*
```

```
enddo
```

```
end
```

2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

Ένας υπολογισμός του π ξανά... (Leibniz)

```
integer :: n=0
```

```
real(8) :: pi = 4 * atan(1.d0), pid = 0, s = 0
```

```
do
```

```
  if(abs(pi - pid) < 1e-14) exit
```

```
  s = s + ((-1)**n / (2 * n + 1.))
```

```
  pid = 4 * s
```

```
  n = n + 1
```

```
enddo
```

```
write(*,10) 'calculated pi=', pid
```

```
write(*, '(a15,f7.5)') 'calculated pi=', pid
```

```
print 10, 'calculated pi=', pid
```

```
print '(a15,f7.5)', 'calculated pi=', pid
```

```
10 format(a15,f7.5)
```

```
end
```

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = \frac{\pi}{4}$$

```
calculated pi=3.14159
```

```
calculated pi=3.14159
```

```
calculated pi=3.14159
```

```
calculated pi=3.14159
```

μορφ/τής	είδος δεδομένων	Παράδειγμα	Αποτέλεσμα
Fw.d	πραγματικοί: w.d αν αριθμός< w προστίθενται προηγούμενα κενά τα '.' και '-' υπολογίζονται σαν χαρακτήρες	αριθμός= F7.2 F8.2 F9.2 F8.3 F9.3 F10.4	-1234.5678 ***** -1234.57 -1234.57 ***** -1234.568 -1234.5680
Ew.d[Ee]	πραγματικοί με εκθέτη	αριθμός= E7.2 E7.2E1 E8.2 E14.8 E12.6E2	-1234.5678 ***** -.12E+4 -.12E+04 -.12345680E+04 -.123457E+04
ENw.d[Ee]	πραγματικοί με εκθέτη, απεικόνισης μηχανικής	αριθμός= EN7.2 EN9.2E1 EN10.3 EN12.5 EN14.7E2	-1234.5678 ***** -1.23E+3 -1.235E+03 -1.23457E+03 -1.2345678E+03

μορφ/τής	είδος δεδομένων	παράδειγμα	αποτέλεσμα
ES .d[Ee]	πραγματικοί με εκθέτη επιστημονικής απεικόνισης	αριθμός= ES7.2 ES9.2E1 ES10.3 ES12.5 ES14.7E2	-1234.5678 ***** -1.23E+3 -1.235E+03 -1.23457E+03 -1.2345678E+03
Gw .d[Ee]	γενικευμένη απεικόνιση	αριθμός= G7.2 G9.2E1 G10.3 G12.5 G14.7E2	-1234.5678 ***** -.12E+4 -.123E+04 -1234.6 -1234.57
Dw .d[De]	πραγματικοί διπλής ακρίβειας	αριθμός= D7.2 D9.2 D10.3 D12.5 D15.8	-1234.5678 ***** -.12D+04 -.123D+04 -.12346D+04 -.12345678D+04

μορφ/τής	είδος δεδομένων	Παράδειγμα	αποτέλεσμα
I <i>w</i> [.m]	ακέραιοι: <i>m.w</i> αν αριθμός< <i>w</i> προστίθενται προηγούμενα κενά, αν αριθμός< <i>m</i> προστίθενται προηγούμενα μηδενικά	αριθμός= I5 I7.5 I5.2 I4	-1234 -1234 -01234 -1234 ****
L <i>w</i>	λογικές τιμές	τιμή= L1 L2 L3 L4 L5	.true. T T T T T
A [<i>w</i>]	Αλφαριθμητικοί	τιμή= A A1 A3 A4 A5	'test' test t tes test test

μορφ/τής	Χρήση	Χρήση με READ	Χρήση με WRITE ή PRINT
' χαρακτήρες '	μεταφέρει τους χαρακτήρες στην μονάδα εξόδου	OXI	NAI
Tc, TLc, TRc	καθορίζει την θέση της στήλης 'c' στο πεδίο	NAI	NAI
nX	μετατοπίζει την θέση στο αρχείο κατά 'n' χαρακτήρες	NAI	NAI
SP, SS, S	εμφάνιση του προαιρετικού '+'	OXI	NAI
/	μετατοπίζει στο επόμενο πεδίο ή σηματοδοτεί το τέλος του πεδίου (αλλάζει γραμμή)	NAI	NAI
:	σηματοδοτεί το τέλος της μορφοποίησης	OXI	NAI
kP	καθορίζει την κλίμακα των εκθετών στους ακολουθούντες επαναλήψιμους μορφοποιητές 'F' και 'E'	NAI	NAI
BN, BZ	καθορίζει τη σημασία των κενών σε αριθμητικά πεδία	NAI	OXI

Επικοινωνία με `namelist`

NAMELIST /όνομα/ λίστα δεδομένων, /όνομα2/.....

- ▶ χρησιμοποιείται στις εντολές εισόδου-εξόδου με τον όρο:

[NML=] *όνομα*

- ▶ Στην είσοδο γράφουμε αρχικά *&όνομα*, και μετά: *όνομα μεταβλητής=τιμή*, .
Αντί για '&', μπορούμε να χρησιμοποιήσουμε '\$'. Στο τέλος γράφουμε '/' ή '&'.

```
integer :: a, b
namelist /test/a,b
  read (*,test)
  write (*,test)
end
```

```
$test a=1 b=2 /
&TEST
A=          1,
B=          2,
/
```

Άνοιγμα και κλείσιμο αρχείων

OPEN, CLOSE

```
OPEN ( [UNIT=] αριθμός [ ,ACCESS=πρόσβαση] [ ,ACTION=ενέργεια]  
[ ,BLANK=κενά] [ ,DELIM=οριοθέτησης] [ ,ERR=ετικέτα σφάλματος]  
[ ,FILE=όνομα] [ ,FORM=φόρμα] [ ,IOSTAT=κατάσταση] [ ,PAD=  
[ ,POSITION=θέση] [ ,RECL=μήκος πεδίου]  
[ ,STATUS=κατάσταση αρχείου] )
```

```
CLOSE ( [UNIT=] αριθμός [ ,ERR=ετικέτα σφάλματος]  
[ ,IOSTAT=κατάσταση] [ ,STATUS=κατάσταση αρχείου] )
```



Παρατηρείτε ότι η μόνη απόλυτα απαραίτητη παράμετρος είναι ο αριθμός της λογικής μονάδας και, στις περισσότερες περιπτώσεις, το όνομα του αρχείου.

Διαβάζω ένα άγνωστο αρχείο

`iostat`

```
program iotest
  real::a; integer::ioerr, i
  open (10, file='data.dat')
  do i = 1, 10
    write(10,*) i
  enddo
  close (10)
  open (10, file='data.dat')
  do
    read(10,*, iostat = ioerr) a
    if (ioerr == 0) then
      print*, a * a
    else
      exit
    end if
  end do
  close (10)
end program iotest
```

```
1.00000000
4.00000000
9.00000000
16.00000000
25.00000000
36.00000000
49.00000000
64.00000000
81.00000000
100.00000000
```

Διαβάζω πάλι ένα άγνωστο αρχείο

`iostat, iomsg`

```
program iotest
  real::a; integer::ioerr,i; character(80)::msg
  open (10,file='data.dat')
  do i=1,10
    write(10,*) i
  enddo
  close (10)
  open (10,file='data.dat')
  do
    read(10,*,iostat=ioerr,iomsg=msg) a
    if (ioerr/=0) then
      print*,msg; exit
    else
      print*,a*a
    end if
  end do
  close (10)
end program iotest
```

```
1.00000000
4.00000000
9.00000000
16.00000000
25.00000000
36.00000000
49.00000000
64.00000000
81.00000000
100.00000000
End of file
```

Διαβάζω ένα άγνωστο αρχείο αλλιώς...

end

```
program iotest
  real::a; integer::ioerr,i
  open (10,file='data.dat')
  do i = 1 , 10
    write(10,*)i
  enddo
  close (10)
  open (10,file='data.dat')
  do
    read(10,*,end=20) a
    print*,a*a
  end do
  20 continue
  close (10)
end program iotest
```

Το εγγενές άρθρωμα iso_fortran_env

```
program iso_env
  use iso_fortran_env
  implicit none
10 format (a25,i5)
  print 10, 'input_unit=', input_unit
  print 10, 'output_unit=', output_unit
  print 10, 'error_unit=', error_unit
  print 10, 'iostat_end=', iostat_end
  print 10, 'iostat_eor=', iostat_eor
  print 10, 'file_storage_size=', file_storage_size
  print 10, 'integer_kinds=', integer_kinds
  print 10, 'real_kinds=', real_kinds
  print 10, 'numeric_storage_size=', numeric_storage_size
  print 10, 'character_storage_size=', character_storage_size
end program iso_env
```


Το εγγενές άρθρωμα `iso_fortran_env`

```
input_unit=      5
output_unit=     6
error_unit=      0
iostat_end=     -1
iostat_eor=     -2
file_storage_size= 8
integer_kinds=  1   2   4   8
real_kinds=     4   8  10  16
numeric_storage_size= 32
character_storage_size= 8
```

```
Process returned 0 (0x0)   execution time : 0.047 s
```

```
program unknown_file
  use iso_fortran_env
  implicit none
!  δηλώσεις:
  real:: x; integer :: err, n = 0
  character(30) :: msg, xfile
!  αρχή:
  first:do
  print '(/, a, /)', 'Give the name of the file or &
                        & <none> to exit'

  read (input_unit, *) xfile
  open (unit = 9, file = xfile, status = 'old', &
        & iostat = err, iomsg = msg)

      if(err /= 0) then
        if (xfile == 'none') exit first
        print '(/,a,/)', msg

      else
```

```
print *, 'Opening ', xfile
second:do
  read (9, *, iostat = err, iomsg = msg) x
  if (err == iostat_end) then
    print *, msg;
    print '(/, a, i3, a, /)', 'I have read', &
      & n, ' numbers'

    exit second
  else
    n = n + 1
    print *, x
  endif
enddo second
close (9)
endif
enddo first
print *, 'successful completion'
end program unknown_file
```

Άγνωστο αρχείο

```
Give the name of the file or <none> to exit
```

```
dat.txt
```

```
File 'dat.txt' does not exist
```

```
Give the name of the file or <none> to exit
```

```
data.dat
```

```
Opening data.dat
```

```
1.00000000
```

```
2.00000000
```

```
3.00000000
```

```
4.00000000
```

```
5.00000000
```

```
6.00000000
```

```
7.00000000
```

```
8.00000000
```

```
9.00000000
```

```
10.00000000
```

```
End of file
```

```
I have read 10 numbers
```

```
Give the name of the file or <none> to exit
```

```
none
```

```
successful completion
```

Τυχαία δεδομένα

Για τυχαίους αριθμούς στο διάστημα $x_1, x_2 \rightarrow r_s = x_1 + r(x_2 - x_1)$

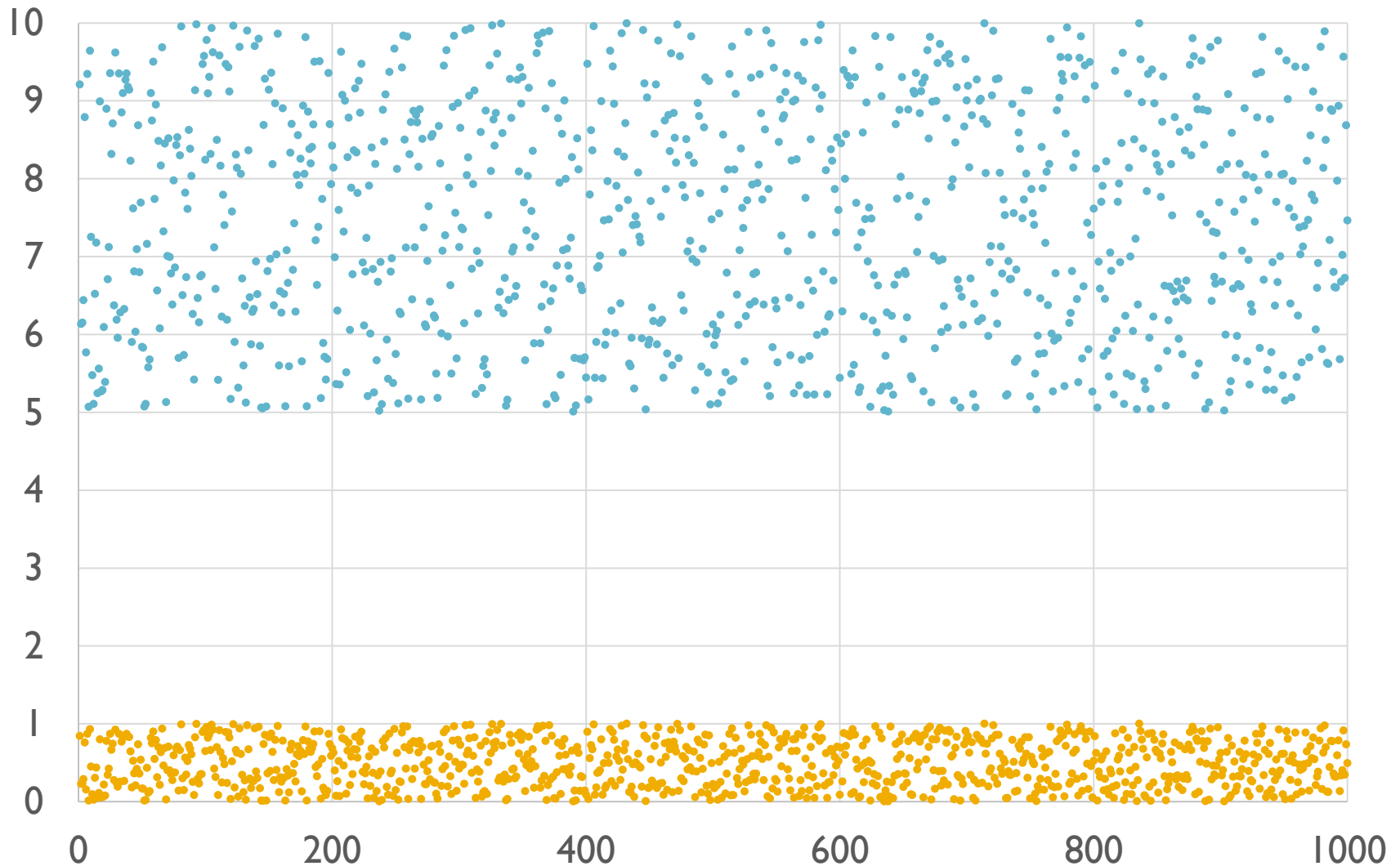
```
program random_data
  implicit none
  ! δηλώσεις:
  integer :: seed(12), counter, i
  real    :: x1 = 5., x2 = 10., rs, r
  ! αρχή:
  call system_clock(counter) !επιστρέφει μια τιμή από
                             !το ρολόι του συστήματος

  seed = counter
  call random_seed(put = seed) !αλλάζει το σημείο εκκίνησης
                               !της γεννήτριας ψευδοτυχαίων

  open(1, file = 'random.txt')
  do i = 1, 1000
    call random_number(r) !επιστρέφει τυχαία τιμή  $0 < r < 1$ 
    rs = x1 + r * (x2 - x1) !στο διάστημα  $x_1 < rs < x_2$ 
    write (1, '(i5, 2x, 2f8.3)') i, r, rs
  end do
  close(1)
end program random_data
```

Τυχαία δεδομένα

Για τυχαίους αριθμούς στο διάστημα $x_1, x_2 \rightarrow r_s = x_1 + r(x_2 - x_1)$



‘Θορυβημένα’ τυχαία δεδομένα

```
program noisy
  implicit none
  !δηλώσεις:
  integer  :: seed(12), counter, x
  real     :: a = 5., b = 10., w = 50., y, yn, r
  !αρχή:
  call system_clock(counter)      !επιστρέφει μια τιμή
                                   !από το ρολόι του συστήματος

  seed = counter
  call random_seed(put = seed)    !αλλάζει το σημείο εκκίνησης
                                   !της γεννήτριας

  open(1, file = 'noisy.dat')
  do x = 1, 100
    y = a + b * x
    call random_number(r)         !επιστρέφει μια τυχαία τιμή
    yn = y + w * (r - 0.5)        !προσθέτει τυχαίο θόρυβο
    write(1, '(i3, 2x, 2f8.3)') x, y, yn
  end do
  close(1)
end program noisy
```

Πως διαβάζουμε το αρχείο στο Excel

The screenshot displays the Microsoft Excel interface. The ribbon is set to the **DATA** tab, which is highlighted with a red border. The ribbon includes the following groups and options:

- FILE**: Standard file operations.
- HOME**: Basic formatting and editing tools.
- INSERT**: Tools for inserting tables, charts, and other elements.
- PAGE LAYOUT**: Tools for adjusting the page layout.
- FORMULAS**: Tools for working with formulas and functions.
- DATA** (highlighted):
 - Get External Data**: From Access, From Web, From Text, From Other Sources, Existing Connections.
 - Connections**: Refresh All, Properties, Edit Links.
 - Sort & Filter**: Sort (A-Z, Z-A), Filter.
- REVIEW**: Tools for proofreading and reviewing.
- VIEW**: Tools for changing the view of the worksheet.

Below the ribbon, the formula bar shows the active cell is **A1**. The worksheet grid shows column **A** selected, with cells **A1** through **A5** visible.

Πως διαβάζουμε το αρχείο στο Excel

The image shows the Microsoft Excel interface with the 'DATA' ribbon selected. The 'From Text' button in the 'Get External Data' group is highlighted with a red box. A green arrow points from this button to the 'noisy.dat' file in the 'Import Text File' dialog box. The dialog box shows the file path: My Documents > OneDrive > workspace > ask 6-16. The file 'noisy.dat' is highlighted with a red box. A green arrow points from the 'noisy.dat' file to the 'All Files' dropdown menu in the 'File name' field. Another green arrow points from the 'noisy.dat' file to the 'Open' button, which is also highlighted with a red box. The 'Tools' button is visible next to the 'Open' button.

Πως διαβάζουμε το αρχείο στο Excel

Text Import Wizard - Step 1 of 3

The Text Wizard has determined that your data is Fixed Width.

If this is correct, choose Next, or choose the data type that best describes your data.

Original data type

Choose the file type that best describes your data:

Delimited - Characters such as commas or tabs separate each field.

Fixed width - Fields are aligned in columns with spaces between each field.

Start import at row: 1 File origin: 737 : Greek (DOS)

My data has headers.

Preview of file F:\Users\Dimitrios\Documents\SkyDrive\workspace\ask 6-16\noisyy.dat

1	0	5.000	-3.850
2	1	15.000	14.641
3	2	25.000	7.101
4	3	35.000	57.264
5	4	45.000	63.918
6	5	55.000	42.385

Cancel < Back **Next >** Finish

Text Import Wizard - Step 2 of 3

This screen lets you set field widths (column breaks).
Lines with arrows signify a column break.

To CREATE a break line, click at the desired position.
To DELETE a break line, double click on the line.
To MOVE a break line, click and drag it.

Data preview

	10	20	30	40	50	60	70	80
0	5.000	-3.850						
1	15.000	14.641						
2	25.000	7.101						
3	35.000	57.264						
4	45.000	63.918						
5	55.000	42.385						

Cancel < Back **Next >** Finish

Text Import Wizard - Step 3 of 3

This screen lets you select each column and set the Data Format.

Column data format

General

Text

Date: DMY

Do not import column (skip)

'General' converts numeric values to numbers, date values to dates, and all remaining values to text.

Advanced...

Data preview

	General	General
0	5.000	-3.850
1	15.000	14.641
2	25.000	7.101
3	35.000	57.264
4	45.000	63.918
5	55.000	42.385

Cancel < Back Next > **Finish**

Πως διαβάζουμε το αρχείο στο Excel

Import Data ? x

Select how you want to view this data in your workbook.

Table
 PivotTable Report
 PivotChart
 Only Create Connection

Where do you want to put the data?

Existing worksheet:

New worksheet

Add this data to the Data Model

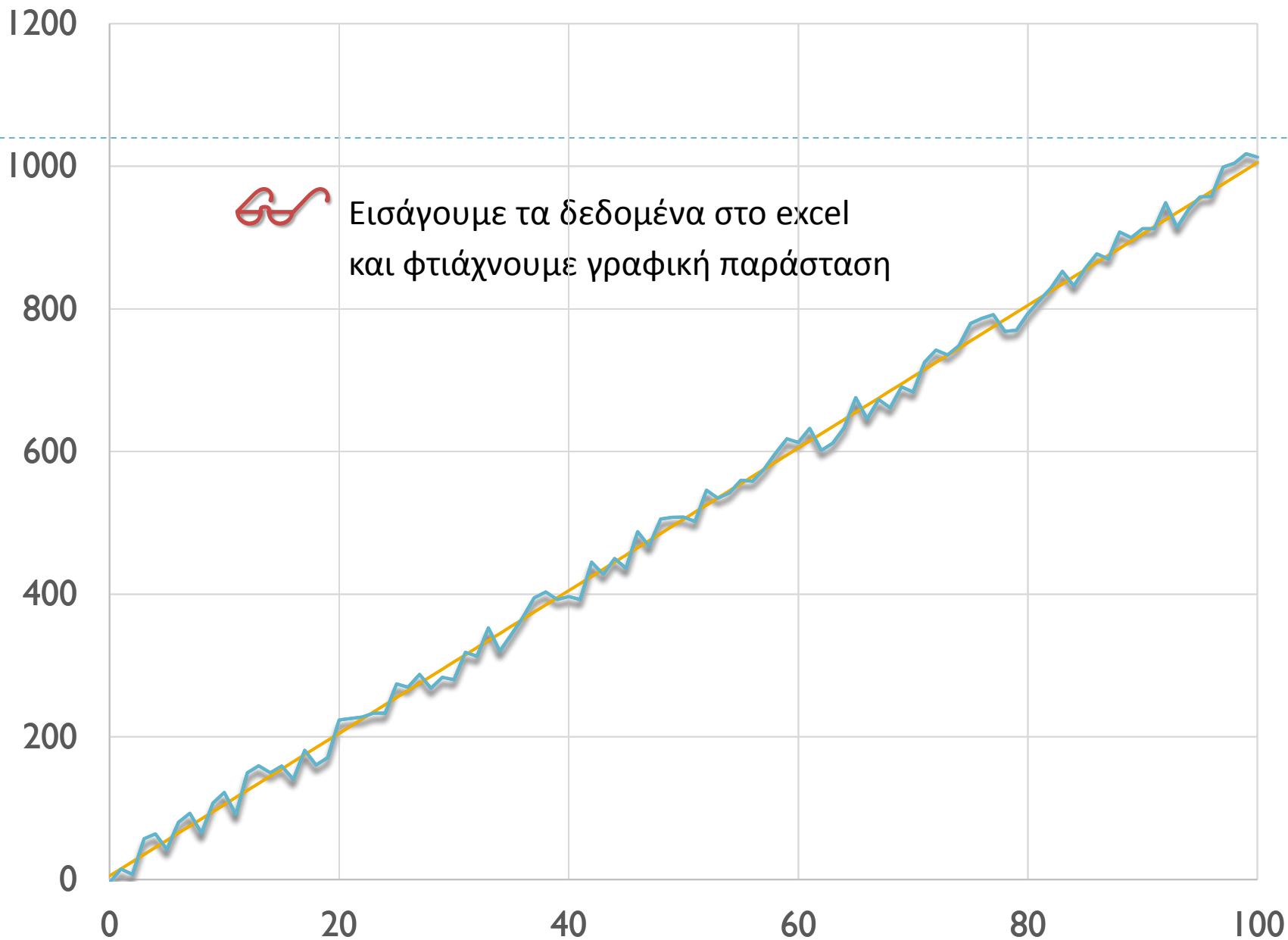
Properties... **OK** Cancel

HOME INSERT PAGE LAYOUT FORMULAS **DATA** REVIEW VIEW

From Text Get External Data From Other Sources Existing Connections Refresh All Connections Properties Edit Links Sort Filter

fx

	A	B	C	D	E	F	G	H	I
1	0	5	-3.85						
2	1	15	14.641						
3	2	25	7.101						
4	3	35	57.264						
5	4	45	63.918						
6	5	55	42.385						
7	6	65	80.323						
8	7	75	92.739						
9	8	85	64.974						
10	9	95	107.121						
11	10	105	122.047						

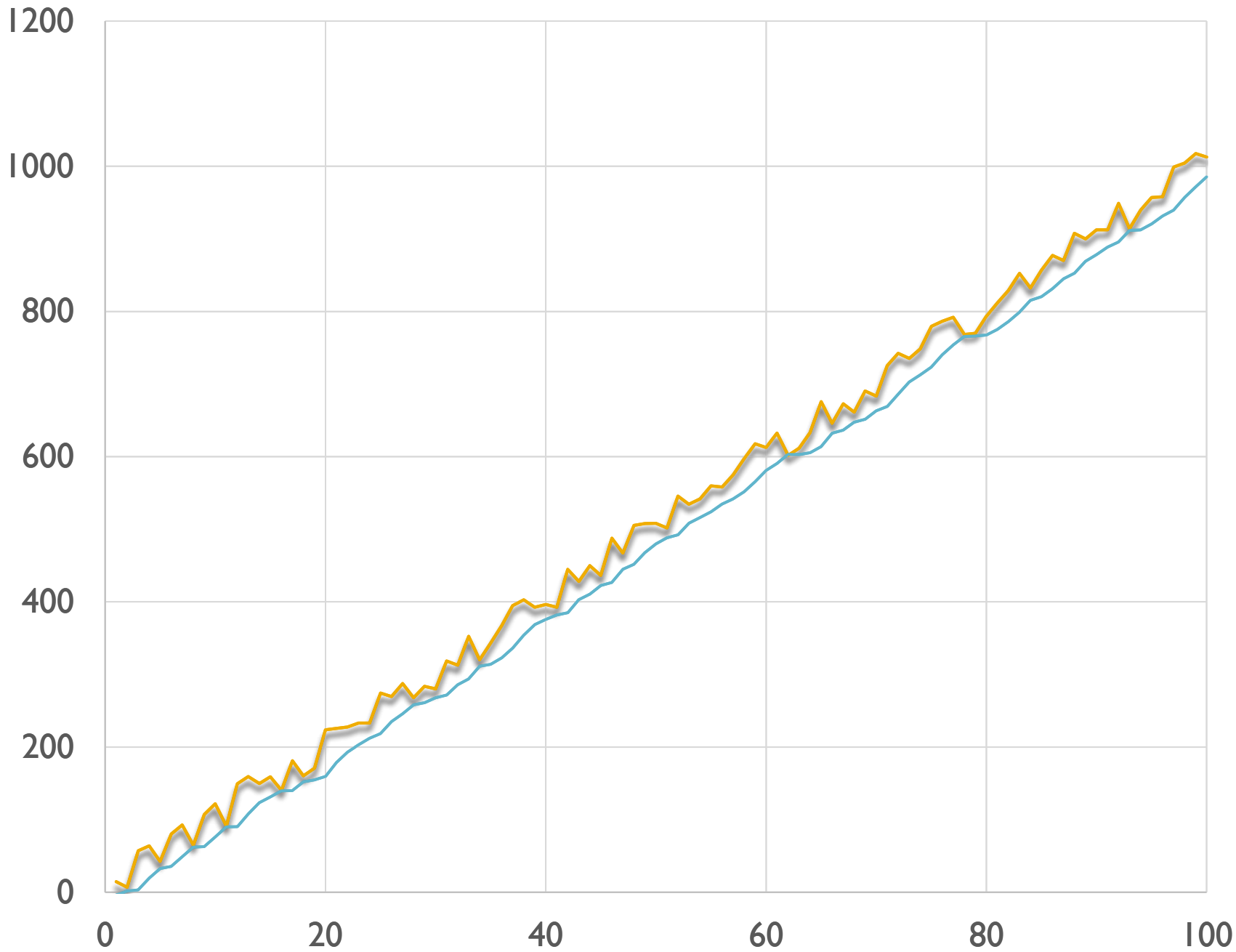


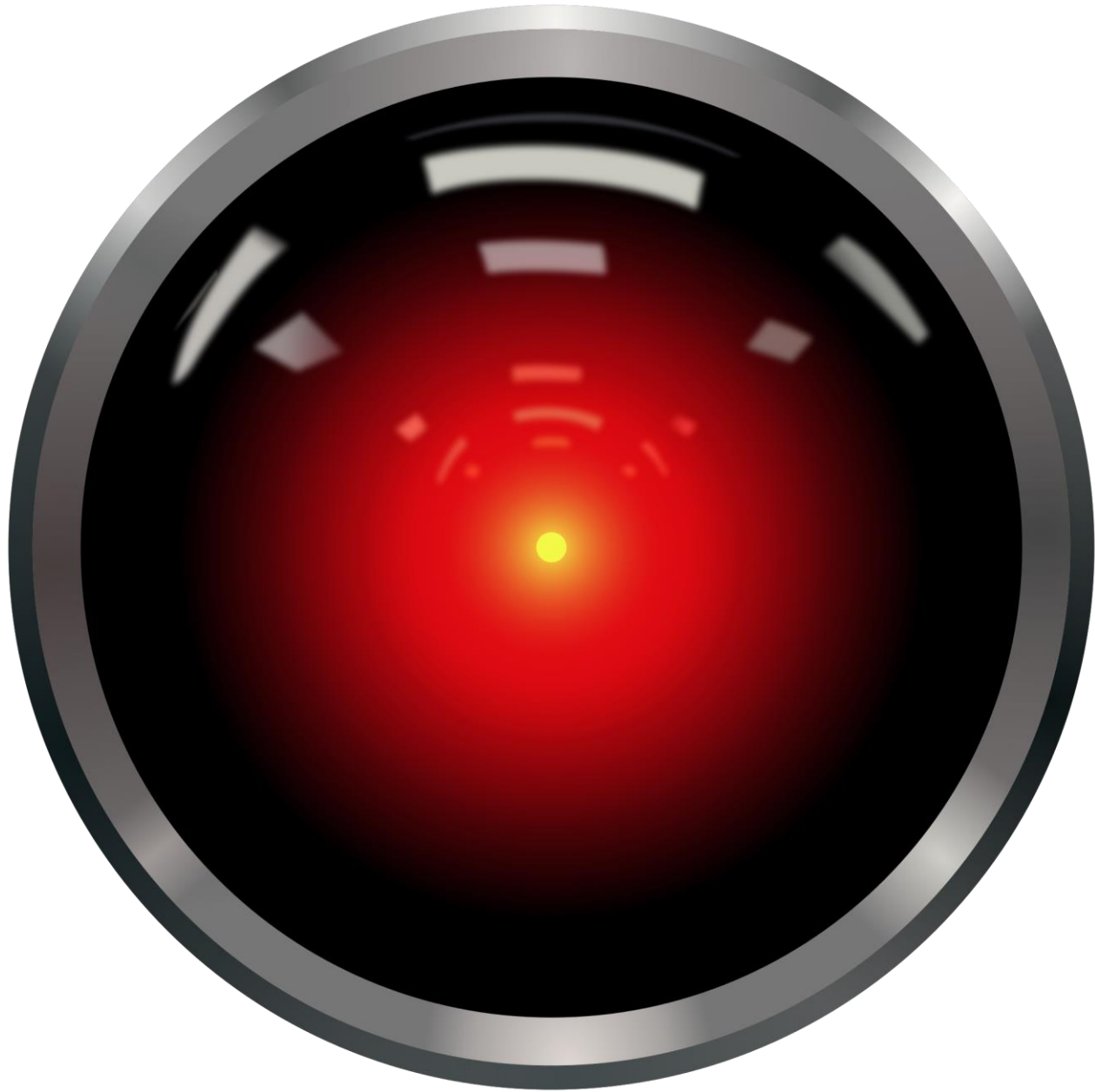
Εκθετική εξομάλυνση

$$y_i = y_{i-1} + w (x_{i-1} - y_{i-1})$$

```
program smoothing
  use iso_fortran_env; implicit none
!  δηλώσεις:
  real      :: y, y_1, x0, x, w = 0.3
  integer   :: i, ioerr = 0

!  αρχή:
  open(1, file = '..\ask 6-16\noisy.dat', status='old')
  open(2, file = 'smoothed.dat')
    read(1, '(i3, 2x, f8.3)') i, x
    y_1 = x
    do while(ioerr /= iostat_end)
      y = y_1 + w * (x - y_1)
      write (2, '(i3, 2x, 2f8.3)') i, x, y
      y_1 = y
      read (1, '(i3, 2x, 2f8.3)', iostat = ioerr) i, x0, x
    end do
  close(1); close(2)
end program smoothing
```





Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Πατρών, Όνομα μέλους ή μελών ΔΕΠ 2014:
Δημήτριος Ματαράς. «Εισαγωγή στον Προγραμματισμό Η/Υ». Έκδοση: 1.0.
Πάτρα 2014. Διαθέσιμο από τη δικτυακή διεύθυνση:
<https://eclass.upatras.gr/courses/CMNG2178>.

Χρηματοδότηση

- ▶ Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στο πλαίσιο του εκπαιδευτικού έργου του διδάσκοντα.
- ▶ Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Αθηνών**» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού.
- ▶ Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.