



# Εισαγωγή στον Προγραμματισμό Η/Υ (Fortran 90/95/2003)

Ενότητα 3: Ο τελεστής ανάθεσης και οι εντολές  
εισόδου – εξόδου

Νίκος Καραμπετάκης  
Τμήμα Μαθηματικών



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» 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.



# Περιεχόμενα Ενότητας

1. Ο τελεστής ανάθεσης = και η βασική του διαφορά από το σύμβολο ισότητας.
2. Η εντολή ανάγνωσης μεταβλητών READ.
3. Η εντολή εκτύπωσης μεταβλητών WRITE.



# Σκοποί Ενότητας

1. Η παρουσίαση του τελεστή ανάθεσης (=).
2. Η παρουσίαση της εντολής εισόδου READ.
3. Η παρουσίαση της εντολής εξόδου WRITE.



# Δομή προβλημάτων που θα ασχοληθούμε

- Χρειαζόμαστε εντολές για είσοδο δεδομένων (READ).
- Χρειαζόμαστε εντολές για επεξεργασία δεδομένων (=).
- Χρειαζόμαστε εντολές για έξοδο δεδομένων (WRITE – PRINT).



# Ο τελεστής ανάθεσης

Μεταβλητή=Έκφραση

**Παράδειγμα.**

A=1

A=A+1

B=A

A	1

A	2



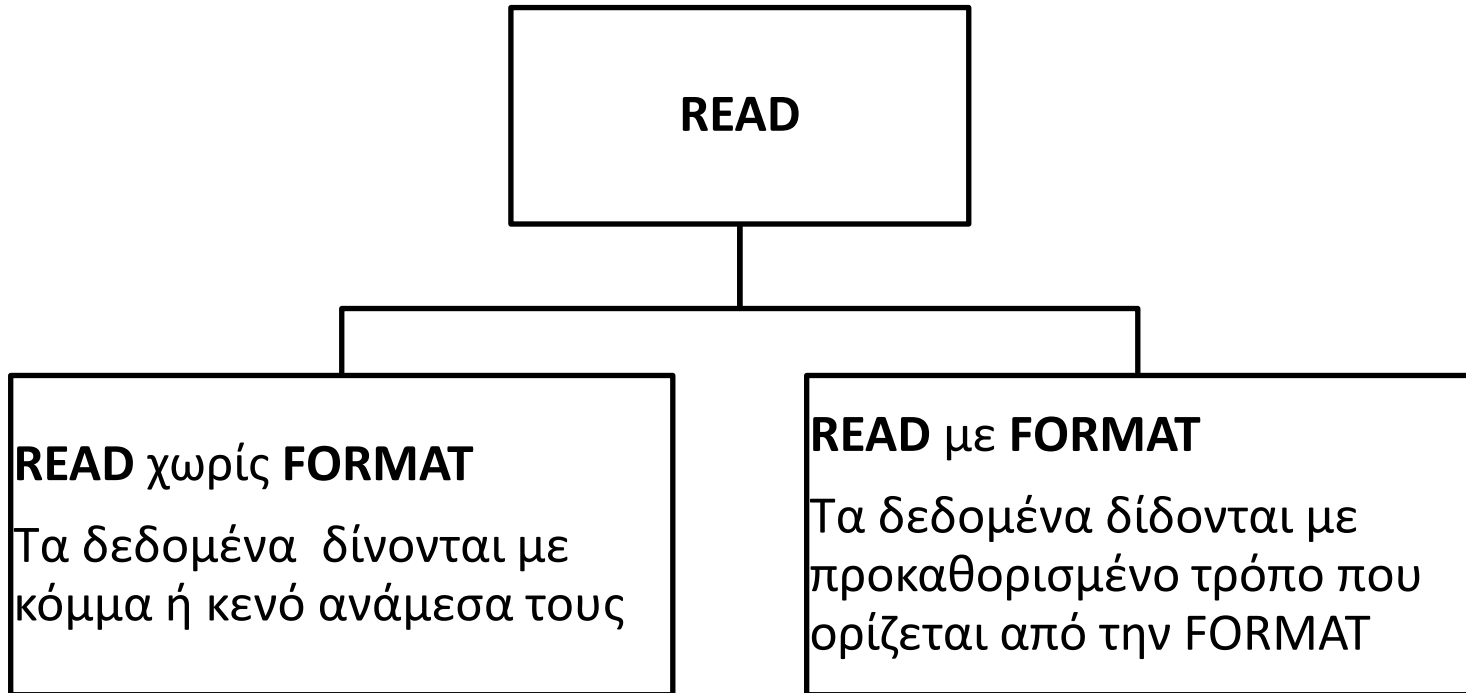
# Παρατηρήσεις

1. Προσοχή κατά την τοποθέτηση των παρενθέσεων.
2. Χρήση του ίδιου ονόματος μεταβλητής σε όλα τα σημεία του προγράμματος.
3. Οι μεταβλητές δεξιά του = έχουν πάρει τιμή.
4. Τα ορίσματα έχουν σωστές τιμές.
5. Ονόματα μεταβλητών σχετικά με το τι παριστούν.
6. Χρήση της PARAMETER για σταθερές.
7. Σπάσιμο πολύπλοκων εκφράσεων σε απλούστερες.
8. Χρήση των ενσωματωμένων συναρτήσεων.





# Η εντολή εισόδου READ



# Η εντολή εισόδου READ χωρίς FORMAT

## Σύνταξη:

READ\*, λίστα μεταβλητών

ή

READ(n,\*) λίστα μεταβλητών

Όπου n=\*,0 ή 5 που δηλώνει ως μονάδα εισόδου το πληκτρολόγιο.

## Παράδειγμα.

**READ\*, A, B**

READ(5,\*) A,B

READ(\*,\*) A,B



# Παράδειγμα READ χωρίς FORMAT

```
PROGRAM TEST
```

```
IMPLICIT NONE      ! Όλες οι μεταβλητές θα δηλώνονται.
```

```
INTEGER :: A,B,Y   ! Δήλωση των ακέραιων μεταβλητών
```

```
READ*,A,B          ! Απαιτεί την εισαγωγή 2 ακεραίων τιμών για A,B
```

```
Y=MOD(A,B)         ! Υπολογίζει το υπόλοιπο της ακεραίας διαίρεσης του A  
                   ! με το B και το τοποθετεί στην μεταβλητή Y
```

```
END PROGRAM TEST
```

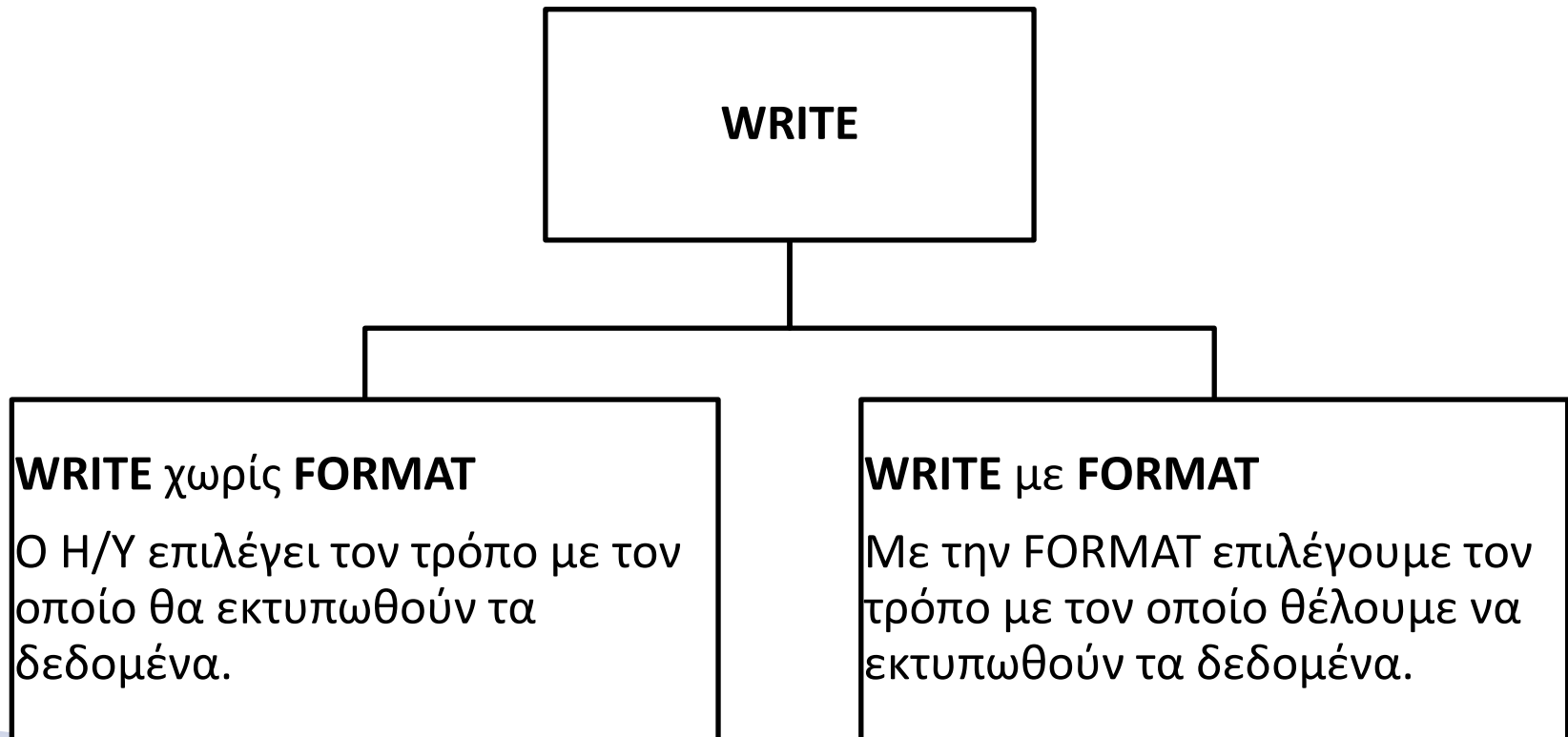


# Παρατηρήσεις

1. Αντιστοίχιση αριθμού μεταβλητών, τύπου δεδομένων, σειράς δεδομένων.
2. Κατά την εισαγωγή δεδομένων διαχωρίζουμε τα δεδομένα με κόμμα και τα strings τα τοποθετούμε σε εισαγωγικά.
3. Τι γίνεται αν τα δεδομένα που τοποθετούμε είναι (περισσότερα, ίσα, λιγότερα) από αυτά που χρειάζεται η READ;
4. Κάθε READ διαβάζει από καινούρια γραμμή.



# Η εντολή εξόδου PRINT – WRITE



# Η εντολή εξόδου WRITE χωρίς FORMAT

## Σύνταξη:

PRINT\*, λίστα {σταθερών, μεταβλητών ή εκφράσεων}

ή

WRITE(n,\*) λίστα {σταθερών, μεταβλητών ή εκφράσεων}

όπου n=\*,0 ή 6 που δηλώνει ως μονάδα εξόδου την οθόνη.

## Παράδειγμα.

```
PRINT*, 'A=',A,'B=', B
```

```
WRITE(6,*) 'A=',A,'B=',B
```

```
WRITE(*,*) 'A=',A,'B=',B
```



# Παράδειγμα PRINT

```
PROGRAM TEST
```

```
IMPLICIT NONE    ! Όλες οι μεταβλητές θα δηλώνονται.
```

```
INTEGER :: A,B,Y ! Δήλωση των ακέραιων μεταβλητών
```

```
PRINT*,'Give me two values for A,B :'
```

```
READ*,A,B        ! Απαιτεί την εισαγωγή 2 ακεραίων τιμών για A,B
```

```
Y=MOD(A,B)       ! Υπολογίζει το υπόλοιπο της ακεραίας διαίρεσης του A  
                 ! με το B και το τοποθετεί στην μεταβλητή Y
```

```
PRINT*,'RESIDUE=','Y
```

```
PRINT*,'QUOTIENT=','(A-Y)/B
```

```
END PROGRAM TEST
```



# Πως θα εμφανίζονται τα αποτελέσματα ;

1. α. οι ακέραιοι αριθμοί θα καταλαμβάνουν 11 θέσεις,  
β. οι πραγματικοί αριθμοί θα καταλαμβάνουν 16 θέσεις, από τις οποίες οι 6 θέσεις θα είναι για τα δεκαδικά ψηφία,  
γ. οι πραγματικοί αριθμοί διπλής ακρίβειας θα καταλαμβάνουν 24 θέσεις, από τις οποίες οι 15 θέσεις θα είναι για τα δεκαδικά ψηφία
2. Μεταξύ των αριθμών που θα εκτυπώνονται θα υπάρχει ένα κενό, ενώ μεταξύ αριθμών και χαρακτήρων δεν υπάρχει κενό.
3. Οι αριθμοί θα στοιχίζονται προς τα δεξιά, σε αντίθεση με τις αλυσίδες χαρακτήρων που θα έχουν αριστερή στοίχιση.
4. Η κάθε WRITE-PRINT εκτυπώνει σε διαφορετική γραμμή.





# Προπαθήστε να μεταφράσετε και να εκτελέσετε το παρακάτω παράδειγμα

```
PROGRAM TEST
  IMPLICIT NONE
  INTEGER (4)          :: A
  REAL (4)             :: B
  REAL (8)             :: C
  CHARACTER (10)      :: NAME
  A=3
  B=45.2
  C=23.56
  NAME='NIKOS'
  PRINT*,A,B,C,NAME
END PROGRAM TEST
```

\* Δυαδική μορφή του 23.56 : 10111.100011110101110000101



# Δραστηριότητα

Τι πιστεύετε ότι θα εκτυπωθεί από το παρακάτω πρόγραμμα ;

```
PROGRAM TEST
  IMPLICIT NONE
  INTEGER      :: LENGTH
  REAL         :: HEIGHT
  CHARACTER    :: NAME
  LENGTH=3 ; HEIGHT=5.2
  NAME='ANASTASIA'
  PRINT*,LENGTH, HEIGHT
  WRITE(6,*) NAME
END PROGRAM TEST
```



# Απλά προβλήματα εισόδου – εξόδου (Περιγραφή)

Καταθέτουμε σε μια τράπεζα στην αρχή του κάθε έτους ένα σταθερό ποσό  $A$ . Αν η κατάθεση γίνεται με επιτόκιο  $E\%$ , τότε το κεφάλαιο  $K$  μετά την νιοστή κατάθεση θα είναι :

$$K = \frac{A * (1 + E) * [(1 + E)^N - 1]}{E}$$

Να γραφεί πρόγραμμα υπολογισμού του  $K$  αν είναι γνωστά τα  $A$ ,  $E$  και  $N$ .

## A) Περιγραφή εισόδου-εξόδου

*Είσοδος* – Το σταθερό ποσό  $A$ , το επιτόκιο  $E\%$  και τα χρόνια  $N$ .

*Έξοδος* – Το συνολικό κεφάλαιο  $K$  μετά από  $N$  χρόνια.



# Απλά προβλήματα εισόδου – εξόδου (Αλγόριθμος)

**Β) Παράδειγμα με το χέρι.**

Έστω  $N=1$  χρόνος,  $E=0.02$  και  $A=1000€$ .

$$K = \frac{1000 * (1 + 0.02) * [(1 + 0.02)^1 - 1]}{0.02}$$

**Γ) Αλγόριθμος προβλήματος**

**Βήμα 1.** Διαβάζουμε τις τιμές α) του αρχικού κεφαλαίου  $A$ , β) του επιτοκίου  $E$ , και γ) των χρόνων  $N$ .

**Βήμα 2.** Υπολογίζουμε το συνολικό κεφάλαιο από τον τύπο που μας δόθηκε.

**Βήμα 3.** Εκτυπώνουμε τα αποτελέσματα.



# Απλά προβλήματα εισόδου – εξόδου (Πρόγραμμα)

## Δ) Πρόγραμμα

### PROGRAM EXAMPLE31

! Το πρόγραμμα αυτό υπολογίζει το τελικό κεφάλαιο K, μετά από  
!ετήσια κατάθεση ποσού A, για N συνεχόμενα χρόνια και με επιτόκιο  
E

! A=ποσό ετήσιας κατάθεσης, E=επιτόκιο, N=χρόνια, K=τελικό ποσό

! Δήλωση σταθερών, μεταβλητών

```
IMPLICIT NONE
```

```
INTEGER      :: N
```

```
REAL         :: E,A,K
```



# Απλά προβλήματα εισόδου – εξόδου (συνέχεια)

! ΒΗΜΑ 1. ΕΙΣΑΓΩΓΗ ΑΡΧΙΚΩΝ ΤΙΜΩΝ

```
PRINT*, 'ΠΟΣΟ ='
```

```
READ*, A
```

```
PRINT*, 'ΡΑΤΕ ='
```

```
READ*, E
```

```
PRINT*, 'ΧΡΗΜΑΤΑ ='
```

```
READ*, N
```

! ΒΗΜΑ 2. ΥΠΟΛΟΓΙΣΜΟΣ ΤΕΛΙΚΟΥ ΠΟΣΟΥ

$$K = A * (1 + E) * ((1 + E) ** N - 1) / E$$

! ΒΗΜΑ 3. ΕΚΤΥΠΩΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

```
PRINT*, 'ΤΕΛΙΚΟ ΠΟΣΟ=','K,' EURO'
```

**END PROGRAM EXAMPLE31**



# Άσκηση 1

Να διαβαστούν οι κάθετες πλευρές  $A$ ,  $B$  ενός ορθογωνίου τριγώνου και στη συνέχεια να υπολογιστεί α) η υποτείνουσα του τριγώνου  $C$ , και β) το εμβαδόν  $E$  του τριγώνου, από τους τύπους :

$$C = \sqrt{A^2 + B^2}, E = \frac{1}{2} A \times B$$



# Η εντολή εισόδου READ με FORMAT

## Σύνταξη:

READ f, λίστα μεταβλητών

ή

READ(n,f) λίστα μεταβλητών

όπου  $n=*,0$  ή 5 που δηλώνει ως μονάδα εισόδου το πληκτρολόγιο,

f=ετικέτα (label) (θετικός ακέραιος αριθμός) που δηλώνει τη θέση της FORMAT που αντιστοιχεί στην συγκεκριμένη READ

f FORMAT(λίστα από περιγραφές)





# Χαρακτηριστικά της FORMAT

Χαρακτηριστικό	Περιγραφή
nX	N κενοί χαρακτήρες
Iw	W θέσεις για ακέραιο αριθμό
Fw.d	W θέσεις για πραγματικό από τις οποίες d θέσεις για δεκαδικά. $\underbrace{\pm xx.\overbrace{xxxx}^d}_{w}$
Ew.d	W θέσεις για πραγματικό κινητής υποδιαστολής από τις οποίες d θέσεις για δεκαδικά. $\underbrace{\pm xx.\overbrace{xx}^d E \pm xx}_{w}$
Dw.d	W θέσεις για πραγματικό διπλής ακρίβειας από τις οποίες d θέσεις για δεκαδικά. $\underbrace{\pm xx.\overbrace{xx}^d D \pm xx}_{w}$



# Χαρακτηριστικά της FORMAT

## ...συνέχεια

Χαρακτηριστικό	Περιγραφή
A[w]	W θέσεις για αλυσίδα χαρακτήρων
Lw	W θέσεις για λογικούς χαρακτήρες



# Παράδειγμα 1

```
PROGRAM TEST
```

```
    IMPLICIT NONE
```

```
    REAL :: X
```

```
    READ 10, X
```

```
10    FORMAT(F9.1)
```

```
    PRINT*,X
```

```
END
```

						2	4	5
					2	4	5	
				2	4	5		
	2	4	.	5				

\* Η υποδιαστολή κατά την είσοδο των δεδομένων έχει μεγαλύτερη ισχύ από τη FORMAT.



# Παράδειγμα 2

```
PROGRAM BIRTHDAY  
  
    IMPLICIT NONE  
  
    INTEGER      :: D,M,Y  
  
    READ(5,10) D,M,Y  
  
10    FORMAT(I2,1X,I2,1X,I4)  
  
    PRINT*,D,'-',M,'-',Y  
  
END
```



# Παράδειγμα 3

```
PROGRAM TEST
  IMPLICIT NONE
  CHARACTER (10) :: NAME
  LOGICAL CONDITION
  READ(5,10) NAME,CONDITION
10  FORMAT(A10,1X,L1)
  PRINT*,NAME,CONDITION
END
```



# Άσκηση 2

Να γραφεί πρόγραμμα που θα διαβάζει τα στοιχεία μιας πιστωτικής κάρτας :

XXXX-XXXX-XXXX-XXXX

και θα εμφανίζει τους 4 αριθμούς της πιστωτικής κάρτας.



# Η εντολή εξόδου WRITE με FORMAT

## Σύνταξη:

PRINT f, λίστα {σταθερών, μεταβλητών ή εκφράσεων}

ή

WRITE(n,f) λίστα {σταθερών, μεταβλητών ή εκφράσεων}

όπου n=\*,0 ή 6 που δηλώνει ως μονάδα εξόδου την οθόνη

και f=το label (ετικέτα) της FORMAT που συνδέεται με την αντίστοιχη WRITE-PRINT.

f FORMAT(λίστα από περιγραφές)



# Επιπλέον χαρακτηριστικά της FORMAT για εκτύπωση

Πρώτος χαρακτήρας FORMAT	Περιγραφή
'1'	Η εκτύπωση της γραμμής θα γίνει σε νέα σελίδα.
Κενό (ή 1X)	Η εκτύπωση της γραμμής θα γίνει στην αμέσως επόμενη γραμμή.
'0'	Η εκτύπωση της γραμμής θα γίνει στη μεθεπόμενη γραμμή.
'+'	Η εκτύπωση της γραμμής θα γίνει στην ίδια γραμμή.

\* Οι υπόλοιποι χαρακτήρες ακολουθούν την ίδια δομή με αυτή της READ.





# Παραδείγματα PRINT και WRITE με FORMAT

```
PRINT 10  
10 FORMAT('1','RESULTS')
```

```
WRITE(6,10)  
10 FORMAT('0','123')
```

```
COMPLEX :: A  
READ*,A  
WRITE(6,10) A  
10 FORMAT(1X,F5.1,1X,F5.1)
```



# Επιπλέον Χαρακτήρες

Επιπλέον Χαρακτήρες	Περιγραφή
<b>String</b>	Εμφανίζει το αντίστοιχο string στην οθόνη.
<b>nH</b>	Μεταφέρει τους επόμενους n χαρακτήρες που ακολουθούν το H στην έξοδο (πρόκειται να καταργηθεί σε επόμενες εκδόσεις).
<b>Tc, Tlc, TRc</b>	Προσδιορίζει τη στήλη στην οποία θα εκτυπωθεί αυτό που θέλουμε : θα εκτυπωθεί στη c στήλη, θα εκτυπωθεί στη c στήλη προς τα αριστερά σε σχέση με αυτήν που βρισκόμαστε τώρα, θα εκτυπωθεί στην c στήλη προς τα δεξιά σε σχέση με αυτήν που βρισκόμαστε τώρα.
<b>/</b>	Αλλάζει γραμμή. Πρέπει να ακολουθεί ο πρώτος χαρακτήρας που προσδιορίζει την κάθετη θέση της επόμενης γραμμής.
<b>\</b>	Συνεχίζει στην ίδια γραμμή.



# Παράδειγμα 4

```
10 WRITE(6,10)  
   FORMAT(1X,TR20,'MAIN RESULTS'/1X,TR10,'X',TR5,4HY(X))
```



# Παρατηρήσεις στην FORMAT

- 1. Πολλές READ/WRITE χρησιμοποιούν την ίδια FORMAT**  
    READ(5,10) A  
    READ(5,10) B  
10 FORMAT(1X,F5.2)
- 2. Μεταξύ READ/WRITE και FORMAT ισχυρότερη είναι η πρώτη**  
    WRITE(6,10) A,B,C  
10 FORMAT(1X,F8.2,F4.1)
- 3. Αποτέλεσμα εκτύπωσης \*\*\* (χρησιμοποίησε Ew.d).**
- 4. Δήλωση του πρώτου χαρακτήρα της FORMAT**  
    WRITE(6,10)  
10 FORMAT('NIKOS')
- 5. Προσθήκη τίτλων στα αποτελέσματα**  
    WRITE(6,10) A  
10 FORMAT(1X,'A=',F5.1,'cm')



# Παράδειγμα 5

..... από προηγούμενο παράδειγμα .....

! ΒΗΜΑ 2. ΥΠΟΛΟΓΙΣΜΟΣ ΤΕΛΙΚΟΥ ΠΟΣΟΥ

$$T=A*(1+E)*((1+E)**N-1)/E$$

! ΒΗΜΑ 3. ΕΚΤΥΠΩΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

```
WRITE(6,10) A,E,N,K
```

```
10FORMAT(1X, 'ΕΤΗΣΙΟ ΠΟΣΟ=' ,F10.2, ' EURO' &  
& /1X, 'ΕΡΙΤΟΚΙΟ=' ,F5.2, &  
& /1X, 'ΕΤΗ=' ,I3, ' ΧΡΟΝΙΑ', &  
& /1X, 'ΤΕΛΙΚΟ ΠΟΣΟ=' ,F10.2, ' EURO')
```

**END PROGRAM EXAMPLE42**



# Άσκηση 2

Να γραφεί πρόγραμμα που θα διαβάζει τα στοιχεία μιας πιστωτικής κάρτας:

XXXX-XXXX-XXXX-XXXX

και θα εμφανίζει τον αριθμό:

XXXXXXXXXXXXXXXXXXXX



# Άσκηση 3

Να διαβαστεί η ακτίνα ενός κύκλου  $R$  και να υπολογιστεί α) η περίμετρος  $P$ , και β) το εμβαδόν  $E$  του κύκλου από τους τύπους :

$$P = 2\pi R, E = \pi R^2$$

όπου  $\pi = 3.14$ .



# Βιβλιογραφία

Ν. Καραμπετάκης, *Εισαγωγή στη Fortran 90/95/2003*, 2<sup>η</sup> Έκδοση, Θεσσαλονίκη: Εκδόσεις Ζήτη, 2011.





# Σημείωμα Αναφοράς

Copyright Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, Καραμπετάκης Νικόλαος. «Εισαγωγή στον Προγραμματισμό Η/Υ (Fortran 90/95/2003). Ο τελεστής ανάθεσης και οι εντολές εισόδου – εξόδου». Έκδοση: 1.0. Θεσσαλονίκη 2014. Διαθέσιμο από τη δικτυακή διεύθυνση: <http://eclass.auth.gr/courses/OCRS145/>.



# Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, 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-sa/4.0/>





# Τέλος Ενότητας

Επεξεργασία: Αναστασία Γ. Γρηγοριάδου

Θεσσαλονίκη, Χειμερινό Εξάμηνο 2013-2014



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ  
2007-2013  
πρόγραμμα για την ανάπτυξη  
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

