



ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΧΤΑ
ΑΚΑΔΗΜΑΪΚΑ
ΜΑΘΗΜΑΤΑ



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

Ενότητα 7: Πολυδιάστατοι Πίνακες

Νίκος Καραμπετάκης
Τμήμα Μαθηματικών



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

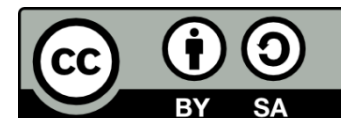


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



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.



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



Περιεχόμενα Ενότητας

1. Χαρακτηριστικά στοιχεία πολυδιάστατων πινάκων.
2. Πως ορίζουμε ένα πολυδιάστατο πίνακα και έναν υποπίνακα.
3. Πως τοποθετούμε τιμές σε ένα πολυδιάστατο πίνακα.
4. Οι εντολές WHERE, FORALL.
5. Επεξεργασία και εμφάνιση τιμών ενός πίνακα.
6. Πράξεις πινάκων.
7. Συναρτήσεις πινάκων.



Σκοποί Ενότητας

1. Η παρουσίαση των πολυδιάστατων πινάκων.
2. Η παρουσίαση των τρόπων εισαγωγής τιμών/ εμφάνισης τιμών σε πολυδιάστατο πίνακα.
3. Η παρουσίαση των πράξεων και συναρτήσεων πολυδιάστατων πινάκων.
4. Η παρουσίαση των μεθόδων επεξεργασίας των στοιχείων ενός πολυδιάστατου πίνακα.



Χαρακτηριστικά στοιχεία πολυδιάστατων πινάκων

- *τάξη*: το πλήθος των διαστάσεων του πίνακα.
- *έκταση*: το πλήθος των στοιχείων τους σε κάθε διάσταση.
- *μορφή*: η τάξη και η έκτασή τους.
- *μέγεθος*: το συνολικό πλήθος των στοιχείων τους.

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}, B_{1ij} = \begin{pmatrix} 1 & 10 \\ 2 & 9 \\ 3 & 8 \end{pmatrix}, B_{2ij} = \begin{pmatrix} 2 & 3 \\ 4 & 9 \\ 1 & 9 \end{pmatrix}$$



Με ποιο τρόπο αποθηκεύεται στην μνήμη του Η/Υ ένας πολυδιάστατος πίνακας ;

..κατά στήλες

C_{11}	C_{12}	C_{13}
C_{21}	C_{22}	C_{23}

....
C_{11}
C_{21}
C_{12}
C_{22}
C_{13}
C_{23}
....



Πως ορίζουμε έναν πολυδιάστατο πίνακα;

Στατικοί πίνακες

```
INTEGER, DIMENSION(3,2) :: A
```

```
INTEGER, DIMENSION(1980:1982,1:2) :: B
```

```
(ή INTEGER A(3,2), B(1980:1982,2))
```

Δυναμικοί πίνακες

```
REAL, DIMENSION(:,:), ALLOCATABLE :: C
```

```
(ή REAL, ALLOCATABLE :: C(:,:))
```

```
READ*,N,M
```

```
ALLOCATE( C(N,M) )
```

```
.....
```

```
DEALLOCATE(C)
```

```
(ή REAL C(:,:))
```



Πως ορίζουμε υποπίνακες;

$A(1:5:2)$! $A(1),A(3),A(5)$

$A(1:5)$! $A(1)$ έως το $A(5)$

$A(3:)$! Από το $A(3)$ έως το τέλος

$A(:)$! Από το $A(1)$ έως το $A(10)$

$A(:5)$! Από το $A(1)$ έως το $A(5)$

$B(1:3,1:3)$! Οι πρώτες 3 γραμμές και 3 πρώτες στήλες του πίνακα B

$B(:,:)$! Όλος ο πίνακας B

$B(3,:)$! Η τρίτη γραμμή του πίνακα B



Πως τοποθετούμε τιμές σε έναν πίνακα; (1ος τρόπος)

```
REAL, DIMENSION(3,2) :: A
```

Ανάγνωση κατά γραμμές

```
READ(*,*) ((A(I,J),J=1,2),I=1,3)
```

Ανάγνωση κατά στήλες

```
READ(*,*) ((A(I,J),I=1,3),J=1,2)
```



Πως τοποθετούμε τιμές σε έναν πίνακα; (2ος τρόπος)

Ανάγνωση κατά γραμμές

```
DO I=1,3
    DO J=1,2
        READ*,A(I,J)
    END DO
END DO
```

Ανάγνωση κατά στήλες

```
DO J=1,2
    DO I=1,3
        READ*,A(I,J)
    END DO
END DO
```



Πως τοποθετούμε τιμές σε έναν πίνακα; (3ος τρόπος)

READ(u2,f) A

όπου

$u2 = *$ ή 0 ή 5

f = είναι ο αριθμός της *format* (αν $f = *$ δεν έχουμε δική μας *format*)



Πως τοποθετούμε τιμές σε έναν πίνακα; (Αν γνωρίζουμε τα στοιχεία του πίνακα)

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$$

DATA A/1,3,5,2,4,6/

ή

A=RESHAPE(SOURCE = (/ 1, 3, 5, 2, 4, 6 /), SHAPE = (/ 3, 2 /))

$$(1 \quad 3 \quad 5 \quad 2 \quad 4 \quad 6) \rightarrow \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$$



Πως τοποθετούμε τιμές σε έναν πίνακα;
(Αν τα στοιχεία του πίνακα ικανοποιούν κάποια ιδιότητα)

$$A = I_4 \rightarrow a_{i,j} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

```
INTEGER, DIMENSION(4,4) :: A
DO I=1,4
  DO J=1,4
    IF (I==J) THEN
      A(I,J)=1
    ELSE
      A(I,J)=0
    END IF
  END DO
END DO
```

END DO



Άσκηση 1

Να δημιουργήσετε τον παρακάτω πίνακα.

$$B = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$



Η εντολή WHERE – Fortran 90

[όνομα :] WHERE (λογική έκφραση 1)

Εντολή ανάθεσης 11

Εντολή ανάθεσης 12

.....

[ELSEWHERE

Εντολή ανάθεσης 01

Εντολή ανάθεσης 02

.....]

END WHERE [όνομα]



Παράδειγμα (Where – Fortran 90)

Να γραφεί μέρος προγράμματος που θα τοποθετεί 1 στα θετικά στοιχεία, -1 στα αρνητικά στοιχεία και 0 στα μηδενικά στοιχεία ενός πίνακα A.

WHERE (A>0) A=1

WHERE (A==0) A=0

WHERE (A<0) A=-1



Η εντολή WHERE – Fortran 95/2003

[όνομα :] WHERE (λογική έκφραση 1)

Εντολή ανάθεσης 11

Εντολή ανάθεσης 12

.....

[ELSEWHERE (λογική έκφραση 2)

Εντολή ανάθεσης 21

Εντολή ανάθεσης 22

.....

[ELSEWHERE

Εντολή ανάθεσης 01

Εντολή ανάθεσης 02

.....]

END WHERE [όνομα]



Παράδειγμα (Where – Fortran 95/2003)

Να γραφεί μέρος προγράμματος που θα τοποθετεί 1 στα θετικά στοιχεία, -1 στα αρνητικά στοιχεία και 0 στα μηδενικά στοιχεία ενός πίνακα A.

```
WHERE (A>0)
```

```
    A=1
```

```
ELSEWHERE
```

```
    WHERE (A==0)
```

```
        A=0
```

```
    ELSEWHERE
```

```
        A=-1
```

```
    END WHERE
```

```
END WHERE
```

ή

```
WHERE (A>0)
```

```
    A=1
```

```
ELSEWHERE (A==0)
```

```
    A=0
```

```
ELSEWHERE
```

```
    A=-1
```

```
END WHERE
```



Η εντολή FORALL

FORALL (μεταβλητή = αρχική τιμή : τελική τιμή : βήμα
[,μεταβλητή = αρχική τιμή : τελική τιμή : βήμα]...) *εντολή
αντικατάστασης*

[όνομα:]**FORALL** (μεταβλητή = αρχική τιμή : τελική τιμή : βήμα
[,μεταβλητή = αρχική τιμή : τελική τιμή : βήμα]...)

Εντολές αντικατάστασης ή Εντολές WHERE ή εντολές FORALL

END FORALL [όνομα]

Παράδειγμα

FORALL (I=1:N,J=1:N) A(I,J)=1/REAL(I+J)



Αν υπάρχουν πάνω από μια δηλώσεις, ο επεξεργαστής θα
τελειώσει με την πρώτη ανάθεση πριν ξεκινήσει με την
δεύτερη!

```
A=[1,2,3,4,5]
FORALL (I=1:N)
    A(I)=0.5*(A(I)+A(I+1))
    B(I)=A(I+1)
END FORALL
PRINT*,A
PRINT*B
```

Έξοδος

```
1.500000 2.500000 3.500000 4.500000 5.000000
2.500000 3.500000 4.500000 5.500000 0.000000
```



Πως εμφανίζουμε τις τιμές ενός πίνακα; (1ος τρόπος)

```
REAL, DIMENSION(3,2) :: A
```

Εκτύπωση κατά γραμμές

```
DO I=1,3
    DO J=1,2
        PRINT*,A(I,J)
    END DO
END DO
```

Εκτύπωση κατά στήλες

```
DO J=1,2
    DO I=1,3
        PRINT*,A(I,J)
    END DO
END DO
```



Πως εμφανίζουμε τις τιμές ενός πίνακα; (2ος τρόπος)

```
REAL, DIMENSION(3,2) :: A
```

Εκτύπωση κατά γραμμές

```
WRITE(*,*) ((A(I,J),J=1,2),I=1,3)
```

Εκτύπωση κατά στήλες

```
WRITE(*,*) ((A(I,J),I=1,3),J=1,2)
```



Πως εμφανίζουμε τις τιμές ενός πίνακα; (3ος τρόπος)

```
REAL, DIMENSION(3,2) :: A
```

```
WRITE(u2,f) A
```

όπου

u2= ή 6*

f=είναι ο αριθμός της format

(αν f= δεν έχουμε δική μας format)*



Πράξεις πινάκων – Πρόσθεση, Αφαίρεση

$$c_{ij} = a_{ij} + b_{ij}$$

```
INTEGER, DIMENSION(:, :), ALLOCATABLE :: A,B,C
```

```
READ*,N,M
```

```
ALLOCATE ( A(N,M), B(N,M) )
```

```
READ*,A,B
```

```
DO I=1,N
```

```
    DO J=1,M
```

```
        C(I,J)=A(I,J)+B(I,J) ! C(I,J)=A(I,J)-B(I,J) για αφαίρεση πινάκων
```

```
    END DO
```

```
END DO
```

ή

C=A+B ! C=A-B για αφαίρεση πινάκων



Πράξεις πινάκων – Πολλαπλασιασμός

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{im}b_{mj} = \sum_{k=1}^m a_{ik}b_{kj}, i = 1, 2, \dots, n, j = 1, 2, \dots, l$$

INTEGER, DIMENSION(:, :), ALLOCATABLE :: A, B, C

READ*, N, M, L

ALLOCATE (A(N, M), B(M, L), C(N, L))

READ*, A, B

DO I=1, N

 DO J=1, M

 C(I, J)=0

 DO K=1, L

 C(I, J)=C(I, J)+A(I, K)+B(K, J)

 END DO

 END DO

END DO

ή C=MATMUL(A, B)



Συναρτήσεις πινάκων

ALL (λογική έκφραση)

Προσδιορίζει αν όλα τα στοιχεία του πίνακα ικανοποιούν τη συγκεκριμένη ιδιότητα που περιγράφεται από τη λογική έκφραση, π.χ. $ALL(A>0)$ θα έχει ως αποτέλεσμα TRUE, αν όλα τα στοιχεία του πίνακα είναι θετικά και FALSE διαφορετικά.

ANY (λογική έκφραση)

Προσδιορίζει αν κάποιο από όλα τα στοιχεία του πίνακα ικανοποιεί τη συγκεκριμένη ιδιότητα που περιγράφεται από τη λογική έκφραση, π.χ. $ANY(A>0)$ θα έχει ως αποτέλεσμα TRUE, αν κάποιο από τα στοιχεία του πίνακα είναι θετικά και FALSE διαφορετικά.



Συναρτήσεις πινάκων (...συνέχεια)

COUNT (λογική έκφραση)

Υπολογίζει το πλήθος των στοιχείων που ικανοποιούν κάποια ιδιότητα που περιγράφεται από την λογική έκφραση, π.χ. `COUNT(A>0)` θα υπολογίσει το πλήθος των θετικών στοιχείων του πίνακα A.

TRANSPOSE (όνομα πίνακα)

Υπολογίζει τον ανάστροφο πίνακα.



Συναρτήσεις πινάκων (Πράξεις)

DOT_PRODUCT (ΠίνακαςA, ΠίνακαςB)	Υπολογίζει το εσωτερικό γινόμενο δύο διανυσμάτων ιδίων διαστάσεων, π.χ. DOT_PRODUCT(A,B) υπολογίζει το εσωτερικό γινόμενο των διανυσμάτων A και B.
SUM (όνομα πίνακα)	Υπολογίζει το άθροισμα των στοιχείων του πίνακα π.χ. SUM(A) υπολογίζει το άθροισμα των στοιχείων του πίνακα A.
PRODUCT (όνομα πίνακα)	Υπολογίζει το γινόμενο όλων των στοιχείων του πίνακα π.χ. PRODUCT(A) υπολογίζει το γινόμενο όλων των στοιχείων του πίνακα A.



Συναρτήσεις πινάκων (Πράξεις) (...συνέχεια)

SUM(A,A>0)	Υπολογίζει το άθροισμα των θετικών στοιχείων του πίνακα A.
PRODUCT(A,A/=0)	Υπολογίζει το γινόμενο των μη μηδενικών στοιχείων του πίνακα A.
SUM(A, dim=1)	Υπολογίζει το άθροισμα των στοιχείων του πίνακα A σε κάθε στήλη.
SUM(A, dim=2)	Υπολογίζει το άθροισμα των στοιχείων του πίνακα A σε κάθε γραμμή.



Παρατηρήσεις

1. Για λόγους οικονομίας μνήμης, πριν από κάθε χρήση ενός πίνακα, θα ρωτάς τον εαυτό σου:
 - α) Χρειάζομαι τα δεδομένα παραπάνω από μια φορά ;
 - β) Πρέπει τα δεδομένα αυτά να αποθηκευτούν πριν τα χρησιμοποιήσω;

Αν η απάντηση είναι «Όχι» και στις δύο ερωτήσεις, θα πρέπει να αντικαταστήσεις τον πίνακα με απλές μεταβλητές.

2. Ως διάσταση του πίνακα θα πρέπει να ορίζεις το μέγιστο αριθμό στοιχείων που μπορείς να χρησιμοποιήσεις.
3. Έλεγξε το δείκτη του πίνακα αν είναι ακέραιος και αν έχει ξεπεράσει τα όρια της διάστασης του πίνακα.
4. Έλεγξε αν συμφωνεί ο δείκτης της DO-END DO με το δείκτη του πίνακα κατά τη διάρκεια της ανακύκλωσης.



Άσκηση 2

Να διαβαστούν τα ονοματεπώνυμα N μαθητών καθώς και οι βαθμοί που είχαν σε M μαθήματα και στη συνέχεια να υπολογιστούν τα εξής:

- α) ο μέσος όρος που είχε κάθε μαθητής,
- β) ο μέσος όρος της τάξης σε κάθε μάθημα,
- γ) ο μέσος όρος της τάξης,
- δ) ο μαθητής(ες) με το μεγαλύτερο μέσο όρο καθώς και ο μαθητής(ες) με το μικρότερο μέσο όρο,
- ε) να ταξινομηθούν οι μαθητές ως προς το μέσο όρο σε φθίνουσα σειρά.



Βιβλιογραφία

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



Σημείωμα Αναφοράς

Copyright Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, Καραμπετάκης Νικόλαος. «Εισαγωγή στον Προγραμματισμό Η/Υ (Fortran 90/95/2003). Πολυδιάστατοι Πίνακες». 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

