



104

Αλγόριθμοι, Προγραμματισμός και Δομές Δεδομένων

Εργαστήριο – Κεφάλαιο 7β & 8α
Δυσδιάστατοι πίνακες & Δείκτες

Διδάσκουσα:
Σοφία Αλμπάνη

Άσκηση 7.4

7.4) Να γραφεί πρόγραμμα σε C που να:

α) δημιουργεί έναν **δισδιάστατο πίνακα** ακεραίων διαστάσεων **6×8**, να τον γεμίζει με **τυχαίους αριθμούς στο διάστημα [0–20]** και να τον εμφανίζει, και

β) να επιτρέπει την **ανταλλαγή δύο γραμμών ή δύο στηλών** του πίνακα.

- Ο χρήστης επιλέγει αν θέλει να γίνει ανταλλαγή γραμμών ή στηλών πληκτρολογώντας:
Enter swap type (1:rows):
 - Αν δώσει 1, γίνεται ανταλλαγή γραμμών.
 - Αν δώσει άλλη τιμή, γίνεται ανταλλαγή στηλών.
- Ο χρήστης εισάγει τους αριθμούς των δύο γραμμών ή στηλών που θέλει να ανταλλάξει (αριθμημένες από 1 έως ROWS ή COLS αντίστοιχα).

Στόχος:

Εξάσκηση στη χρήση **δισδιάστατων πινάκων**, στη **διαχείριση γραμμών και στηλών**, και στη χρήση **τυχαίων αριθμών** και **δομών επανάληψης** για επεξεργασία δεδομένων πίνακα.

Άσκηση 7.5

7.5) Να γραφεί πρόγραμμα σε C το οποίο υπολογίζει το **προϊόν δύο πινάκων**.

α) Να διαβαστεί από τον χρήστη ο πίνακας **a**, διαστάσεων **2×3** και ο πίνακας **b**, διαστάσεων **3×2**.

β) Το πρόγραμμα πρέπει να υπολογίσει τον πίνακα **c**, όπου: **c = a × b** (Δηλαδή ο **c** θα έχει διαστάσεις **2×2**).

- Ο υπολογισμός να γίνει με τον τύπο πολλαπλασιασμού πινάκων:

$$c[i][j] = \sum_{k=0}^{M-1} a[i][k] \cdot b[k][j]$$

- Να εμφανίζεται ο πίνακας **c** μορφοποιημένος.

Στόχος:

Εξάσκηση σε **πολλαπλασιασμό πινάκων**, χρήση **τρών εμφωλευμένων βρόχων**, διαχείριση δισδιάστατων πινάκων διαφορετικών διαστάσεων, μαθηματικές πράξεις σε δομές δεδομένων.



Άσκηση 8.1

8.1) Να γραφεί πρόγραμμα σε C που εισάγει δύο ακέραιους αριθμούς και χρησιμοποιεί **δείκτες (pointers)** για να εμφανίσει:

- i. Τις **διευθύνσεις μνήμης** των δύο μεταβλητών.
- ii. Τις **τιμές** που περιέχουν οι μεταβλητές, μέσω αποαναφοράς των δεικτών (χρήση *ptr).
- iii. Τις **διευθύνσεις μνήμης των ίδιων των δεικτών**.

Στόχος:

Η άσκηση στοχεύει στην κατανόηση της σχέσης ανάμεσα σε: μεταβλητές, διευθύνσεις μνήμης, δείκτες, και αποαναφορά τιμών μέσω δεικτών.



Άσκηση 8.2

8.2) Να γραφεί πρόγραμμα σε C που:

- Δηλώνει τρεις ακέραιες μεταβλητές και τρεις δείκτες σε ακέραιο.
- Κάνει τους δείκτες να «δείχνουν» στις αντίστοιχες μεταβλητές.
- Τροποποιεί τις μεταβλητές έμμεσα μέσω των δεικτών, χρησιμοποιώντας πράξεις με αποαναφορά (dereferencing).

Στο τέλος εμφανίζει τις τελικές τιμές των μεταβλητών μέσω των δεικτών.

Στόχος:

Άσκηση στη χρήση δεικτών και στην πρόσβαση τιμών μέσω δεικτών.

Άσκηση 8.3

8.3) Να γραφεί πρόγραμμα σε C που:

- Δηλώνει έναν ακέραιο και έναν πραγματικό αριθμό (double), καθώς και δύο δείκτες αντίστοιχου τύπου.
- Κάνει τους δείκτες να δείχνουν στις μεταβλητές.
- Αναθέτει την τιμή της πραγματικής μεταβλητής στον ακέραιο μέσω των δεικτών.
- Εμφανίζει:
 - τη νέα τιμή του ακεραίου,
 - το μέγεθος (σε bytes) ενός δείκτη σε int,
 - το μέγεθος ενός δείκτη σε double,
 - και το μέγεθος του τύπου double.

Στόχος:

Κατανόηση ανάθεσης τιμών μέσω δεικτών και του μεγέθους των τύπων/δεικτών.

Συνοπτικά

Κατηγορία	Συντακτικό	Περιγραφή
Δήλωση 2D πίνακα	<code>int arr[R][C];</code>	Δημιουργεί πίνακα με R γραμμές και C στήλες.
Πρόσβαση στοιχείου	<code>arr[i][j]</code>	Τιμή στη γραμμή i και στήλη j.
Εισαγωγή τιμών	<code>scanf("%d", &arr[i][j]);</code>	Διάβασμα τιμής σε συγκεκριμένη θέση.
Δήλωση δείκτη	<code>int *ptr;</code>	Μεταβλητή που αποθηκεύει διεύθυνση μνήμης.
Ανάθεση διεύθυνσης	<code>ptr = &x;</code>	Ο δείκτης δείχνει στη διεύθυνση της μεταβλητής x.
Αποαναφορά	<code>*ptr</code>	Πρόσβαση / αλλαγή στην τιμή μεταβλητής που δείχνει ο δείκτης.
Διεύθυνση μεταβλητής	<code>&x</code>	Παίρνουμε τη διεύθυνση μνήμης του x.
Μέγεθος pointer	<code>sizeof(ptr)</code>	Το μέγεθος ενός δείκτη (ίδιο για όλους τους τύπους).
Pointer σε πίνακα	<code>ptr = arr;</code>	Ο ptr δείχνει στο πρώτο στοιχείο του πίνακα.
Μετακίνηση δείκτη	<code>ptr++, ptr+=2</code>	Δείχνει στο επόμενο/μεθεπόμενο στοιχείο του πίνακα.
Πρόσβαση με arithmetic	<code>*(arr + i)</code>	Ισοδύναμο με <code>arr[i]</code> .
Εισαγωγή μέσω pointer	<code>scanf("%f", ptr);</code>	Αποθήκευση τιμής στη θέση όπου δείχνει ο δείκτης.