

## 3η ΣΕΙΡΑ ΑΣΚΗΣΕΩΝ

ΕΡΓΑΣΤΗΡΙΑΚΗΣ ΕΞΑΣΚΗΣΗΣ ΣΤΙΣ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΕΣ ΤΕΧΝΙΚΕΣ

### B' Εξαμήνου ΗΜ-ΜΥ

#### 1. Παράσταση πίνακα ειδικής μορφής

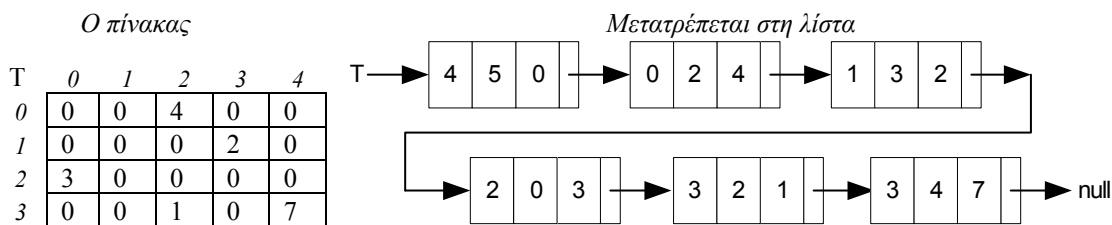
Δίνεται ένας διδιάστατος πίνακας  $\mathbf{u}$  μεγέθους  $N \times N$  (όπου  $N$  θετικός φυσικός αριθμός), του οποίου τα μη μηδενικά στοιχεία βρίσκονται στις θέσεις “ίδιου χρώματος” αν υποθεθεί ότι ο πίνακας έχει τη μορφή σκακιέρας (βλ. σχήμα). Για εξοικονόμηση μνήμης, είναι επιθυμητό να μετατραπεί αυτός ο πίνακας σε ένα μονοδιάστατο πίνακα  $\mathbf{c}$  που να περιέχει μόνο τα μη μηδενικά στοιχεία του  $\mathbf{u}$ . Φυσικά η μετατροπή θα πρέπει να γίνεται κατά τέτοιο τρόπο ώστε να είναι αντιστρέψιμη.

3	0	7	0	1	0	6	0
0	3	0	4	0	9	0	4
5	0	8	0	2	0	3	0
0	6	0	1	0	8	0	1
9	0	8	0	9	0	2	0
0	2	0	5	0	3	0	6
3	0	4	0	6	0	7	0
0	8	0	1	0	4	0	2

- α. Να υπολογίσετε το πλήθος των μη μηδενικών στοιχείων του  $\mathbf{u}$ , ως συνάρτηση του  $N$ .
- β. Να βρείτε μια κατάλληλη συνάρτηση  $\text{loc}(N, i, j)$  που να απεικονίζει μονοσήμαντα τις συντεταγμένες ενός στοιχείου του πίνακα  $\mathbf{u}$  στη συντεταγμένη ενός στοιχείου του πίνακα  $\mathbf{c}$ . Στη συνέχεια, να γράψετε αυτή τη συνάρτηση σε C.
- γ. Να γράψετε δύο συναρτήσεις  $\text{utoc}$  και  $\text{ctou}$  που να μετατρέπουν τον πίνακα  $\mathbf{u}$  στον πίνακα  $\mathbf{c}$  και αντίστροφα. Οι συναρτήσεις αυτές θα πρέπει να δέχονται τους δύο πίνακες ως παραμέτρους.
- δ. Να γράψετε ένα πρόγραμμα που να επιδεικνύει τα παραπάνω, εφαρμόζοντας αυτά σε κατάλληλα δεδομένα της αρεσκείας σας.

#### 2. Παράσταση αραιού πίνακα με λίστα

Ένας πίνακας  $n \times m$  χαρακτηρίζεται ως *αραιός* αν τα περισσότερα στοιχεία του είναι μηδενικά και μπορεί να παρασταθεί με τη βοήθεια μιας απλά συνδεδεμένης λίστας, της οποίας τα στοιχεία περιέχουν πληροφορίες για τη θέση και την τιμή των στοιχείων του αρχικού πίνακα όπως στο παράδειγμα που ακολουθεί.



Να κατασκευάσετε ένα πρόγραμμα που μετατρέπει έναν αραιό πίνακα ακεραίων διαστάσεων  $n \times m$  σε συνδεδεμένη λίστα, κάθε στοιχείο της οποίας έχει πληροφορία για τις συντεταγμένες  $x$  και  $y$  και την τιμή *data* κάθε μη μηδενικού στοιχείου του αρχικού πίνακα. Το πρώτο στοιχείο της λίστας να περιέχει ως συντεταγμένες τις διαστάσεις του αρχικού πίνακα και ως τιμή το 0 (βλ. Σχήμα). Το πρόγραμμά σας θα πρέπει να δέχεται ως είσοδο έναν πίνακα  $\mathbf{T}$ , να εμφανίζει τα στοιχεία του στην οθόνη, να δημιουργεί τη λίστα, να την εμφανίζει στην οθόνη και τέλος να κάνει την αντίστροφη μετατροπή, δηλαδή να δημιουργεί τον πίνακα από τη λίστα και να εμφανίζει τα στοιχεία του στην οθόνη.

#### 3. Παράσταση πολυωνύμων με λίστα

Ένα πολυώνυμο  $A_n x^n + A_{n-1} x^{n-1} + \dots + A_1 x + A_0$  βαθμού  $n$ , μπορεί να παρασταθεί με μια συνδεδεμένη λίστα κάθε στοιχείο της οποίας περιέχει τον εκθέτη του  $m$ -οστού όρου του πολυωνύμου ( $m \leq n$ ) και τον συντελεστή  $A_m$ .

- α. Φτιάξτε ένα πρόγραμμα το οποίο διαβάζει από το πληκτρολόγιο τους όρους ενός πολυωνύμου  $A_n x^n + A_{n-1} x^{n-1} + \dots + A_1 x + A_0$  και ακολούθως δημιουργεί και εμφανίζει στην οθόνη την αντίστοιχη συνδεδεμένη λίστα (επιλέξτε εσείς αν θα συμπεριλάβετε στη λίστα τους όρους με μηδενικό συντελεστή).
- β. Κατασκευάστε μια συνάρτηση που ελέγχει την εγκυρότητα της παράστασης πολυωνύμου με λίστα, επιβεβαιώνοντας ότι η λίστα είναι διατεταγμένη κατά φθίνουσα τάξη ως προς τους εκθέτες.
- γ. Κατασκευάστε μια συνάρτηση η οποία δέχεται ως παράμετρο έναν δείκτη στην κορυφή μιας λίστας που παριστάνει πολυώνυμο και μια πραγματική τιμή  $x$ , και αποτιμά το πολυώνυμο.

δ. Κατασκευάστε μια συνάρτηση η οποία κάνει πρόσθεση πολυωνύμων χρησιμοποιώντας την παράσταση της λίστας.

ε. (ΔΥΣΚΟΛΟ) Κατασκευάστε δύο συναρτήσεις οι οποίες κάνουν πολλαπλασιασμό και διαίρεση πολυωνύμων χρησιμοποιώντας την παράσταση της λίστας.

#### 4. Αύξηση όλων των στοιχείων μιας συνδεδεμένης λίστας

Γράψτε μια συνάρτηση **incList** που να δέχεται ως παράμετρο μια απλά συνδεδεμένη λίστα ακεραίων αριθμών **L** και να παράγει ως αποτέλεσμα μια *άλλη* λίστα ίσου μήκους, της οποίας κάθε στοιχείο να είναι κατά ένα μεγαλύτερο από το αντίστοιχο στοιχείο της **L**.

Παράδειγμα:

**L** : [1, 2, 3, 4, 5]

**incList(L)** : [2, 3, 4, 5, 6]

Γράψτε ένα πρόγραμμα που να επιδεικνύει τη συνάρτηση **incList** εφαρμόζοντάς την σε κατάλληλα δεδομένα της αρεσκείας σας.

#### 5. Εφαρμογή συνάρτησης στα στοιχεία συνδεδεμένης λίστας

Γράψτε μια συνάρτηση **mapList** που να γενικεύει την προηγούμενη, εφαρμόζοντας μια αυθαίρετη συνάρτηση **f** πάνω στα στοιχεία της λίστας **L**. Η συνάρτηση **f** θα πρέπει να περνά ως παράμετρος της **mapList**.

Παράδειγμα:

```
int twice (int x) { return 2*x; }
```

**L** : [1, 2, 3, 4, 5]

**mapList(L, twice)** : [2, 4, 6, 8, 10]

Γράψτε ένα πρόγραμμα που να επιδεικνύει τη συνάρτηση **mapList** εφαρμόζοντάς την σε κατάλληλα δεδομένα της αρεσκείας σας.

#### 6. Φωλιασμένες και επίπεδες λίστες

α. Υλοποιήστε στη γλώσσα C τον τύπο **NestedList** των φωλιασμένων λιστών ακεραίων αριθμών. Τα στοιχεία μιας συνδεδεμένης λίστας τύπου **NestedList** μπορούν να είναι είτε ακέραιοι αριθμοί, είτε άλλες λίστες του ίδιου τύπου.

Παράδειγμα:

**L** : [7, [1, 42, 3], 6, [12, [2, 7], 14]]

β. Γράψτε μια συνάρτηση **flatten** που να δέχεται ως παράμετρο μια φωλιασμένη λίστα ακεραίων αριθμών **L** και να επιστρέφει μια “επίπεδη” λίστα, που να περιέχει τα ίδια στοιχεία με την **L** και με την ίδια σειρά.

Παράδειγμα:

**flatten(L)** : [7, 1, 42, 3, 6, 12, 2, 7, 14]

γ. Γράψτε ένα πρόγραμμα που να επιδεικνύει τη συνάρτηση **flatten**, εφαρμόζοντάς τη σε κατάλληλα δεδομένα της αρεσκείας σας.

#### 7. Εναλλαγή στοιχείων από αρχή και τέλος ακολουθίας

Γράψτε ένα πρόγραμμα που να διαβάζει από ένα αρχείο **INFILE** μια ακολουθία ακεραίων αριθμών, της οποίας το μήκος δεν είναι γνωστό. Στη συνέχεια, να τυπώνει στο αρχείο **OUTFILE** την ίδια ακολουθία εναλλάσσοντας τα στοιχεία της από την αρχή και από το τέλος. Δηλαδή, πρώτα πρέπει να τυπώνεται ο πρώτος ακεραίος, στη συνέχεια ο τελευταίος, έπειτα ο δεύτερος, στη συνέχεια ο προτελευταίος, κ.ο.κ.

Σημείωση: Το αρχείο εισόδου **INFILE** θα πρέπει να διαβάζεται ακριβώς μια φορά. (Αρα μην προσπαθήσετε να βρείτε πρώτα το μήκος της ακολουθίας και μετά να αποθηκεύσετε τα στοιχεία της.)