

ΕΠΑΝΑΛΗΨΗ ΠΙΝΑΚΕΣ Β ΛΥΚΕΙΟΥ

Πολλά από τα προβλήματα που έχουμε αντιμετωπίσει μέχρι στιγμής απλά επεξεργάζονται μία σειρά δεδομένων. Διαβάζουν ένα δεδομένο κάθε φορά, το εκχωρούν σε μία μεταβλητή, εκτελούν τους αντίστοιχους υπολογισμούς και στη συνέχεια επαναλαμβάνεται η ίδια διαδικασία μέχρι να τελειώσουν όλα τα δεδομένα.

Για παράδειγμα ένα τμήμα αλγορίθμου το οποίο διαβάζει τις θερμοκρασίες όλων των ημερών του μήνα, έστω 30, και υπολογίζει τη μέση θερμοκρασία, μπορεί πολύ απλά να γραφεί ως εξής

```
sum ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 30
    ΔΙΑΒΑΣΕ Θ
    sum ← sum + Θ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΜΟ ← sum/30
```

Αυτό που παρατηρούμε, βέβαια, είναι ότι με τη χρησιμοποίηση μόνο μίας μεταβλητής για την αποθήκευση 30 διαδοχικών τιμών θερμοκρασίας (στο συγκεκριμένο παράδειγμα), ουσιαστικά οι τιμές "χάνονται" μετά από κάθε επανάληψη. Η μόνη τελικά θερμοκρασία που "μένει" στη μεταβλητή Θ είναι αυτή της τριακοστής ημέρας, γιατί δε θα εκτελεστεί άλλη επανάληψη. Παρόλα αυτά, η δουλειά μας γίνεται χρησιμοποιώντας μόνο μία μεταβλητή, τη μεταβλητή Θ, το πρόβλημα λύνεται πολύ απλά και το αντίστοιχο πρόγραμμα είναι σύντομο και κατανοητό και πιο "εύκολο" για τον υπολογιστή να το εκτελέσει.

Αν όμως το πρόβλημα άλλαζε και ζητούσε και το πλήθος των ημερών που η θερμοκρασία ήταν κατώτερη της μέσης, τότε η σύγκριση αυτή θα έπρεπε να γίνει μετά τον υπολογισμό της μέσης θερμοκρασίας. Αυτό σημαίνει ότι όλες οι θερμοκρασίες θα έπρεπε να ξαναδιαβαστούν για να συγκριθούν με τη μέση.

Η καλύτερη λύση στο πρόβλημα αυτό είναι η χρήση μεταβλητής με δείκτες, έννοια που είναι γνωστή από τα μαθηματικά και υλοποιείται στον προγραμματισμό με τη δομή δεδομένων του πίνακα. Χρησιμοποιείται λοιπόν μόνο ένα όνομα Θ[], που αναφέρεται και στις τριάντα διαφορετικές θερμοκρασίες.

Το όνομα του πίνακα καθορίζει μία ομάδα διαδοχικών θέσεων στη μνήμη. Κάθε συγκεκριμένη θέση μνήμης καλείται στοιχείο του πίνακα και προσδιορίζεται από την τιμή ενός δείκτη για τους μονοδιάστατους.

Ουσιαστικά οι πίνακες είναι μια ομάδα μεταβλητών που έχουν το ίδιο όνομα και χρησιμοποιούμε, και κάποιους δείκτες (ακέραιους αριθμούς), για να δείξουμε ποια από όλες αυτές τις μεταβλητές θέλουμε. Έτσι τα στοιχεία ενός πίνακα θα τα επεξεργαζόμαστε σαν να είναι μεταβλητές. Θα τους δίνουμε αρχική τιμή όπου χρειάζεται, θα εκχωρούμε σ' αυτά νέες τιμές, θα τα χρησιμοποιούμε σε εκφράσεις κ.ο.κ. Αν ένα στοιχείο ενός πίνακα δεν έχει πάρει καμία τιμή, τότε η τιμή του θα είναι απροσδιόριστη, όπως ακριβώς συμβαίνει και με τις μεταβλητές.

Ένας μονοδιάστατος πίνακας με n στοιχεία (το n σταθερό) μπορεί να αναπαρασταθεί οπτικά με δύο μορφές: α) ως πίνακας – γραμμή και β) ως πίνακας – στήλη. Ουσιαστικά δεν έχει καμία διαφορά για τα προγράμματά μας το πως θα γίνει η οπτικοποίηση.

Στην περίπτωση που θέλουμε να προσπελάσουμε συνολικά όλα τα στοιχεία ενός μονοδιάστατου πίνακα, τότε αρκεί να "πλαισιώσουμε" με μία [ΓΙΑ i από 1 μέχρι n] την εντολή ή τις εντολές θέλουμε να εκτελεστούν για τον πίνακα n θέσεων

"Διάβασμα" των στοιχείων του πίνακα A	Αρχικοποίηση όλων των στοιχείων του A στο 0	Εμφάνιση όλων των στοιχείων του πίνακα A	Υπολογισμός & εμφάνιση πλήθους άρτιων στοιχείων ενός πίνακα A
Για i από 1 μέχρι 5 Διάβασε A[i] Τέλος_επανάληψης	Για i από 1 μέχρι 5 A[i] ← 0 Τέλος_επανάληψης	Για i από 1 μέχρι 5 Γράψε A[i] Τέλος_επανάληψης	pi ← 0 Για i από 1 μέχρι 5 Αν A[i] mod 2 = 0 τότε pi ← pi+1 Τέλος_αν Τέλος_επανάληψης Γράψε "Άρτιοι= ", pi
Για i από 1 μέχρι 5 Διάβασε B A[i] ← B Τέλος_επανάληψης			

Υπολογισμός αθροίσματος – Μ.Ο. μονοδιάστατου πίνακα

Για να βρούμε το άθροισμα των στοιχείων ενός μονοδιάστατου πίνακα, εργαζόμαστε με τη γνωστή μεθοδολογία υπολογισμού αθροίσματος μιας σειράς δεδομένων. Ως γνωστόν, η εντολή υπολογισμού αθροίσματος μίας σειράς αθροιστέων είναι $sum \leftarrow sum + \text{αθροιστέος}$. Εφόσον ο πίνακας είναι στατική δομή δεδομένων, έχει σταθερό πλήθος στοιχείων. Έτσι, σε έναν πίνακα `table ( )` με n στοιχεία το άθροισμα των στοιχείων του `sum` θα ισούται με $table(1) + table(2) + table(3) + \dots + table(n-1) + table(n)$. Χρησιμοποιώντας μία εντολή ΓΙΑ i από 1 μέχρι n ... Τέλος_επανάληψης που "πλαισιώνει" τον γενικό τύπο του αθροίσματος $sum \leftarrow sum + table[i]$ σχηματίζεται το ζητούμενο άθροισμα. Κατόπιν, ο υπολογισμός του ΜΟ είναι εύκολη υπόθεση (Ο πίνακας θεωρείται "γεμάτος" από το προηγούμενο τμήμα του αλγόριθμου):

$sum \leftarrow 0$

Για i από 1 μέχρι n

$sum \leftarrow sum + table[i]$

Τέλος_επανάληψης

$MO \leftarrow sum / n$

Εύρεση min και max και θέσεων τους σε μονοδιάστατο πίνακα

Στην περίπτωση που ζητείται μέγιστο ή ελάχιστο στοιχείο μονοδιάστατου πίνακα, πάλι επεκτείνουμε τη μεθοδολογία που είχαμε μάθει. Συγκρίνουμε διαδοχικά, όλα τα στοιχεία του πίνακα με το `max` ή το `min`, άρα πάλι "πλαισιώνουμε" με ΓΙΑ. ΠΡΟΣΟΧΗ στην αρχικοποίηση των `min` και `max`! Θεωρούμε και πάλι την γενικευμένη περίπτωση ενός μονοδιάστατου πίνακα `table ( )` με n στοιχεία (Ο πίνακας θεωρείται "γεμάτος" από το προηγούμενο τμήμα του αλγόριθμου)

$min \leftarrow table[1]$; θέση $min \leftarrow 1$	!ΑΡΧΙΚΟΠΟΙΗΣΗ MIN
$max \leftarrow table[1]$; θέση $max \leftarrow 1$	!ΑΡΧΙΚΟΠΟΙΗΣΗ MAX!
Για i από 2 μέχρι n	
Αν $table[i] < min$ τότε	!ΕΥΡΕΣΗ MIN!
$min \leftarrow table[i]$	
θέση $min \leftarrow i$	
Τέλος_αν	!ΕΞΩΡΙΣΤΕΣ ΑΝ!
Αν $table[i] > max$ τότε	!ΕΥΡΕΣΗ MAX!
$max \leftarrow table[i]$	
θέση $max \leftarrow i$	
Τέλος_αν	
Τέλος_επανάληψης	

ΤΑΞΙΝΟΜΗΣΗ

Ταξινόμηση (sorting) καλείται η διαδικασία κατά την οποία οι κόμβοι μιας δομής διατάσσονται κατά αύξουσα ή φθίνουσα σειρά.

Υπάρχουν πολλοί διαφορετικοί αλγόριθμοι ταξινόμησης, άλλοι πιο γρήγοροι και άλλοι πιο αργοί. Εμείς, στα πλαίσια του μαθήματος, μαθαίνουμε τον αλγόριθμο της ταξινόμησης ευθείας ανταλλαγής ή φυσαλίδας (bubble sort) όπως είναι γνωστότερος.

Αύξουσα ταξινόμηση
Για i από 2 μέχρι n
Για j από n μέχρι i με_βήμα -1
Αν $table[j-1] > table[j]$ τότε
αντιμετάθεσε $table[j-1], table[j]$
Τέλος_αν
Τέλος_επανάληψης
Τέλος_επανάληψης

Παράλληλοι πίνακες

Στους πίνακες έχουμε αδυναμία να αποθηκεύσουμε διαφορετικού τύπου δεδομένα. Πολλές φορές όμως χρειάζεται να συσχετίζουμε π.χ. ονόματα (χαρακτήρες) με αριθμούς, για παράδειγμα ένα τηλεφωνικός κατάλογος. Για να προσπεράσουμε αυτήν την αδυναμία των πινάκων, χρησιμοποιούμε του λεγόμενους παράλληλους πίνακες. Είναι δύο πίνακες ακριβώς του ίδιου μεγέθους για τους οποίους υπάρχει αντιστοιχία των δεδομένων τους ανά θέση.

Παράδειγμα παράλληλων πινάκων

Σε κάποιους αγώνες ακοντισμού, οι καλύτερες επιδόσεις των 8 αθλητών που παίρνουν μέρος καταχωρούνται στον πίνακα ΕΠΙΔ. Στον πίνακα NAME είναι καταχωρημένα τα ονόματα των αθλητών. Οι πίνακες αυτοί είναι φτιαγμένοι έτσι ώστε η επίδοση και το όνομα ενός αθλητή να βρίσκονται σε θέσεις με τον ίδιο δείκτη στους δυο πίνακες.

Να γίνει αλγόριθμος που θα εμφανίζει το όνομα του αθλητή που παίρνει το χρυσό μετάλλιο.

Λύση:

Οι δύο πίνακες που χρησιμοποιούνται στον συγκεκριμένο αλγόριθμο είναι παράλληλοι. Αυτό που πρέπει να κάνουμε είναι να βρούμε την μέγιστη επίδοση και σε ποια θέση βρίσκεται. Άρα, θα βρούμε το μέγιστο στοιχείο του πίνακα ΕΠΙΔ και σε ποια θέση βρίσκεται, ώστε να εμφανίσουμε το όνομα του αθλητή που βρίσκεται στην ίδια θέση στον πίνακα NAME.

Αλγόριθμος Ακοντισμός

$\text{max} \leftarrow \text{ΕΠΙΔ}[1]$

$\text{θέσηmax} \leftarrow 1$

Για i από 1 μέχρι 8

Αν $\text{ΕΠΙΔ}[i] > \text{max}$ τότε

$\text{max} \leftarrow \text{ΕΠΙΔ}[i]$

$\text{θέσηmax} \leftarrow i$

Τέλος_αν

Τέλος_επανάληψης

Εμφάνισε 'Το χρυσό μετάλλιο παίρνει ο', $\text{NAME}[\text{θέσηmax}]$

Τέλος Ακοντισμός