

Περιεχόμενα

- Γενικά
- Αποθήκευση πινάκων
- Προσπέλαση πινάκων
- Ειδικές μορφές πινάκων
- Πίνακες εγγραφών

Γενικά

- ❖ Η πιο απλή μορφή δομής δεδομένων είναι ο πίνακας, που αναφέρεται σε ένα σύνολο στοιχείων του ίδιου τύπου (real, integer, etc.).
- ❖ Χαρακτηριστικό των πινάκων είναι ότι τα στοιχεία τους αποθηκεύονται σε συνεχόμενες θέσεις μνήμης.
- ❖ Κατά τον ορισμό των πινάκων πρέπει να ορίζουμε την διάσταση του πίνακα για να κρατηθεί ο απαραίτητος χώρος μνήμης.
- ❖ Ο δείκτης που δηλώνει τη θέση του πίνακα είναι μη αρνητικός αριθμός.

Αποθήκευση πινάκων

❖ Έστω ο πίνακας

`int a[lowerbound...upperbound]`

❖ Η θέση του j -στού στοιχείου του πίνακα εκφράζεται ως

$$\text{loc}(a[j]) = \text{loc}(a[\text{lowerbound}]) + j - \text{lowerbound}$$

❖ Αυτό ισχύει όταν το κάθε στοιχείο του πίνακα καταλαμβάνει μία λέξη της κύριας μνήμης. Αν κάθε στοιχείο έχει μήκος s λέξεων τότε η συνθήκη γίνεται

$$\text{loc}(a[j]) = \text{loc}(a[\text{lowerbound}]) + s * (j - \text{lowerbound})$$

Αποθήκευση πινάκων

❖ Υπάρχουν δύο μέθοδοι αποθήκευσης δισδιάστατων πινάκων : Κατά γραμμή (λεξικογραφική διάταξη – lexicographical order) και κατά στήλη (αντίστροφη λεξικογραφική διάταξη – reverse lexicographical order).

❖ Αν ο πίνακας έχει αποθηκευτεί κατά στήλες τότε η θέση του στοιχείου $a[i,j]$ ενός πίνακα a ($l \times m$) είναι:

$$\text{loc}(a[i,j]) = \text{loc}(a[1,1]) + (i-1) + l * (j-1)$$

❖ Αν ο πίνακας έχει αποθηκευτεί κατά γραμμές τότε η θέση του στοιχείου $a[i,j]$ ενός πίνακα a ($l \times m$) είναι:

$$\text{loc}(a[i,j]) = \text{loc}(a[1,1]) + (j-1) + m * (i-1)$$

Αποθήκευση πινάκων

Για τρισδιάστατο πίνακα a: lxmxn αν γεμίσει ως

```
for i=1 to l
  for j=1 to m
    for k=1 to n
      a[i,j,k] = 7;
    end
  end
end
```

τότε

$$\text{loc}(a[i,j,k]) = \text{loc}(a[1,1,1]) + m*n*(i-1) + n*(j-1) + (k-1)$$

Αποθήκευση πινάκων

Οι τρεις τελευταίες εξισώσεις ονομάζονται **συναρτήσεις απεικόνισης (mapping functions)**.

Προσπέλαση πινάκων

❖ Έστω ο πίνακας $a[1..2, 1..4]$ έχει αποθηκευτεί σε θέσεις μνήμης όπως φαίνεται παρακάτω:

$a[1,1]$ $a[1,2]$ $a[1,3]$ $a[1,4]$ $a[2,1]$

2000 2001 2002 2003 2004

Δηλαδή ισχύει: $1999+j$ και $2003+j$

Επομένως αν δημιουργηθεί ένας πίνακας $access[1..2]$ με στοιχεία τις τιμές **1999** και **2003**, τότε η θέση του στοιχείου $a[i,j]$ θα είναι $access[i]+j$.

Προσπέλαση πινάκων

❖ Με τη μέθοδο αυτή εξαλείφονται οι πολλαπλασιασμοί αλλά απαιτείται περισσότερος χώρος μνήμης λόγω ύπαρξης του νέου διανύσματος. Συγκεκριμένα ο πίνακας $a[1..m, 1..n]$, απαιτεί τώρα:

$m+m*n=m*(n+1)$ θέσεις για την αποθήκευσή του

❖ Από τη σχέση αυτή συνεπάγεται ότι ο πίνακας προσπέλασης είναι μικρότερος αν προηγείται η μικρή διάσταση ($m < n$). Αυτό σημαίνει ότι ο πίνακας $a[1..m, 1..n]$ είναι καλύτερα να αποθηκεύεται ως $a[1..n, 1..m]$ αν $m \gg n$.

Ειδικές μορφές πινάκων

❖ Αν ο πίνακας έχει κάποια ειδική μορφή τότε είναι δυνατόν να χρησιμοποιηθεί κάποια ιδιαίτερη τεχνική για την ελάττωση του χρόνου αποθήκευσης του.

Συμμετρικοί και τριγωνικοί πίνακες

Συμμετρικός (symmetric) λέγεται ο τετραγωνικός πίνακας με στοιχεία $a[i,j]=a[j,i]$, $1 \leq i,j \leq n$.

Μπορεί να αποθηκευτεί με την ακόλουθη διάταξη των στοιχείων: $a[1,1]$, $a[2,1]$, $a[2,2]$, .. $a[n,1]$, $a[n,2]$,... $a[n,n]$. Έτσι απαιτούνται $1+2+3+..+n=n(n+1)/2$ θέσεις μνήμης, οπότε η συνάρτηση απεικόνισης είναι

$$\text{loc}(a[i,j]) = \text{loc}(a[1,1]) + i*(i-1)/2 + (j-1)$$

Ειδικές μορφές πινάκων

- ❖ Οι τριγωνικοί πίνακες είναι επίσης τετραγωνικοί πίνακες και διακρίνονται σε άνω (upper – $a[i,j]=0 \ i<j$) και κάτω (lower – $a[i,j]=0 \ i>j$) τριγωνικούς (triangulars). Προφανώς και ισχύουν οι ίδιες παρατηρήσεις με τους συμμετρικούς, όμως οι γλώσσες προγραμματισμού δεν επιτρέπουν τον ορισμό τέτοιων πινάκων, οπότε θα πρέπει να οριστεί ένας μονοδιάστατος πίνακας διαστάσεων $1..n(n+1)/2$, οπότε το στοιχείο $a[i,j]$ θα αντιστοιχεί στο στοιχείο $[i*(i-1)/2+j]$
- ❖ Δύο $n \times n$ τριγωνικοί πίνακες μπορούν να αποθηκευθούν μαζί σε έναν πίνακα $n \times (n+1)$

$$c = \begin{Bmatrix} a_{11} & b_{11} & b_{21} & b_{31} \\ a_{21} & a_{22} & b_{22} & b_{32} \\ a_{31} & a_{32} & a_{33} & b_{33} \end{Bmatrix}$$

Ειδικές μορφές πινάκων

Τριδιαγώνιοι πίνακες

Πρόκειται για τετραγωνικούς πίνακες που έχουν όλα τα στοιχεία τους μηδέν εκτός από τα στοιχεία της κυρίας διαγωνίου και των δύο πλαϊνών.

Ορίζουμε έναν πίνακα διαστάσεων $1..3*n-2$, οπότε το στοιχείο $a[i,j] \rightarrow c[3*i-2]$. Αποθηκεύεται ως $a_{11}, a_{12}, a_{21}, a_{22}, a_{23}, a_{32}, \dots, a_{n(n-1)}, a_{nn}$.

$$\text{loc}(a[i,j]) = \text{loc}(a[1,1]) + (3i-2) + (j-1)$$

Ειδικές μορφές πινάκων

Αραιοί πίνακες

Ένας πίνακας λέγεται αραιός (sparse) όταν ένα μεγάλο ποσοστό των στοιχείων του έχουν τιμή 0. Συνήθως περισσότερα από το 80% των στοιχείων.

1^η μέθοδος αποθήκευσης:

Αν ο αρχικός πίνακας έχει $m \times n$ ακέραια στοιχεία, τότε για την αποθήκευσή του χρειάζονται $m \times n$ λέξεις. Αν ο πίνακας έχει N διάφορα του μηδενός στοιχεία και η λέξη αποτελείται από W bits τότε ο νέος τρόπος αποθήκευσης απαιτεί

$$\begin{Bmatrix} 6 & 0 \\ 0 & 0 \end{Bmatrix} \dashrightarrow \begin{Bmatrix} 1 & 0 \\ 0 & 0 \end{Bmatrix} \& \{6\}$$

$N + m \times n / W$ λέξεις

Ειδικές μορφές πινάκων

2^η μέθοδος αποθήκευσης:

Κάθε μη μηδενικό στοιχείο αποθηκεύεται ως μία τριάδα αριθμών, οι πρώτοι δύο δείχνουν τη θέση του στοιχείου και ο τρίτος την τιμή του στοιχείου.

Ειδικές μορφές πινάκων

Πίνακες εγγραφών

Στην περίπτωση που μία δομή απαιτεί μη ακέραια πολλαπλάσια του μήκους λέξης της μνήμης, δημιουργούνται κενά τα οποία ονομάζονται **παραγεμίσματα (pads)**. Παράγοντας **χρησιμοποίησης μνήμης (memory utilization factor)** ορίζεται ο λόγος

$$u = \frac{s}{\lceil s \rceil}$$

(s: αριθμός λέξεων της εγγραφής)

Μία λύση είναι η τοποθέτηση δύο ή περισσότερων πεδίων σε μία λέξη κάτι που ονομάζεται **συμπύκνωση (packing)**.