

## Κεφάλαιο 3,9

1. Να δοθεί ο ορισμός της δομής δεδομένων.

**Δομή Δεδομένων** είναι ένα σύνολο αποθηκευμένων δεδομένων που υφίστανται επεξεργασία από ένα σύνολο λειτουργιών. (σ.56)

2. Ποιες είναι οι βασικές πράξεις επί των δομών δεδομένων;

Οι βασικές λειτουργίες (ή αλλιώς πράξεις) επί των δομών δεδομένων είναι οι ακόλουθες:

- |   |  |
|---|--|
| <b>Εγγραφή</b><br><br><b>Αρχείο</b><br><br><b>Βάση Δεδομένων</b>  | • <b>Προσπέλαση</b> (access), πρόσβαση σε ένα κόμβο με σκοπό να εξετασθεί ή να τροποποιηθεί το περιεχόμενό του.  |
|   | • <b>Εισαγωγή</b> (insertion), δηλαδή η προσθήκη νέων κόμβων σε μία υπάρχουσα δομή.  |
|   | • <b>Διαγραφή</b> (deletion), που αποτελεί το αντίστροφο της εισαγωγής, δηλαδή ένας κόμβος αφαιρείται από μία δομή.  |
|   | • <b>Αναζήτηση</b> (searching), κατά την οποία προσπελούνται οι κόμβοι μιας δομής, προκειμένου να εντοπιστούν ένας ή περισσότεροι που έχουν μια δεδομένη ιδιότητα. |
|   | • <b>Ταξινόμηση</b> (sorting), όπου οι κόμβοι μιας δομής διατάσσονται κατά αύξουσα ή φθίνουσα σειρά.   |
|   | • <b>Αντιγραφή</b> (copying), κατά την οποία όλοι οι κόμβοι ή μερικοί από τους κόμβους μιας δομής αντιγράφονται σε μία άλλη δομή.                                  |
| • <b>Συγχώνευση</b> (merging), κατά την οποία δύο ή περισσότερες δομές συνενώνονται σε μία ενιαία δομή. |  |
| • <b>Διαχωρισμός</b> (separation), που αποτελεί την αντίστροφη πράξη της συγχώνευσης. (σ56)             |  |

3. Ποια είναι η εξάρτηση μεταξύ της δομής δεδομένων και του αλγορίθμου που επεξεργάζεται τη δομή;

Μία δομή δεδομένων να είναι αποδοτικότερη από μία άλλη δομή με κριτήριο κάποια λειτουργία, για παράδειγμα την αναζήτηση, αλλά λιγότερο αποδοτική για κάποια άλλη λειτουργία, για παράδειγμα την εισαγωγή. Αυτές οι παρατηρήσεις εξηγούν αφ' ενός την ύπαρξη διαφορετικών δομών, και αφ' ετέρου τη σπουδαιότητα της επιλογής της κατάλληλης δομής κάθε φορά. Υπάρχει μεγάλη εξάρτηση μεταξύ της δομής δεδομένων και του αλγορίθμου που επεξεργάζεται τη δομή. Μάλιστα, το πρόγραμμα πρέπει να θεωρεί τη δομή δεδομένων και τον αλγόριθμο ως μία αδιάσπαστη ενότητα. (*Αλγόριθμοι + Δομές Δεδομένων = Προγράμματα* Wirth) (σ.57)

4. Να περιγραφούν οι δύο κυριότερες κατηγορίες των δομών δεδομένων.

Οι δομές δεδομένων διακρίνονται σε δύο κατηγορίες: τις *στατικές* και τις *δυναμικές*.

**Στατικές** : - το ακριβές μέγεθος της απαιτούμενης κύριας μνήμης καθορίζεται κατά τη στιγμή του προγραμματισμού τους, και κατά συνέπεια κατά τη στιγμή της μετάφρασης

- τα στοιχεία αποθηκεύονται σε συνεχόμενες θέσεις μνήμης

**Δυναμικές** : - δεν έχουν σταθερό μέγεθος, αλλά ο αριθμός των κόμβων τους μεγαλώνει και μικραίνει καθώς στη δομή εισάγονται ή διαγράφονται δεδομένα αντίστοιχα κατά τη διάρκεια της εκτέλεσης τους προγράμματος  
- Δεν αποθηκεύονται σε συνεχόμενες θέσεις μνήμης αλλά στηρίζονται στην τεχνική της λεγόμενης δυναμικής παραχώρησης μνήμης. (σ.57)

5. Να περιγραφεί η δομή του πίνακα και να δοθεί παράδειγμα χρήσης του.

Πίνακας είναι μια στατική δομή που περιέχει στοιχεία του ίδιου τύπου (δηλαδή ακέραιους, πραγματικούς κ.λπ).

Η χρήση των στοιχείων ενός πίνακα γίνεται με τη χρήση του συμβολικού ονόματος του πίνακα ακολουθούμενου από την τιμή ενός ή περισσότερων δεικτών (indexes) σε παρένθεση ή αγκύλη. (π.χ. πίνακας[γραμμή, στήλη]). Το πλήθος των στοιχείων του πίνακα ορίζεται κατά τη διάρκεια του προγραμματισμού και οι θέσεις μνήμης είναι συνεχόμενες. (σ.58)

6. Να δοθεί ο ορισμός της στοίβας.

Στοίβα ονομάζεται η δομή δεδομένων στην οποία τα δεδομένα που βρίσκονται στην κορυφή της στοίβας λαμβάνονται πρώτα, ενώ αυτά που βρίσκονται στο βάθος της στοίβας λαμβάνονται τελευταία. Αυτή η μέθοδος επεξεργασίας ονομάζεται *Τελευταίο μέσα, πρώτο έξω* (LIFO Last-In- First -Out). (σ.60)

7. Ποιες είναι οι βασικές λειτουργίες που γίνονται σε μία στοίβα;

Οι κύριες λειτουργίες σε μία στοίβα είναι:

- η ώθηση (push) στοιχείου στην κορυφή της στοίβας, και
- η απώθηση (pop) στοιχείου από τη στοίβα. (σ60)

8. Πώς μπορεί να υλοποιηθεί μια στοίβα και τι έλεγχοι πρέπει να γίνονται;

Μια στοίβα μπορεί να υλοποιηθεί πολύ εύκολα με τη βοήθεια ενός μονοδιάστατου πίνακα. Μια βοηθητική μεταβλητή (με όνομα συνήθως top) χρησιμοποιείται για να δείχνει το στοιχείο που τοποθετήθηκε τελευταίο στην κορυφή της στοίβας.

Η διαδικασία της ώθησης πρέπει οπωσδήποτε να ελέγχει, αν η στοίβα είναι γεμάτη, οπότε λέγεται ότι συμβαίνει *υπερχείλιση* (overflow) της στοίβας ενώ η διαδικασία απώθησης ελέγχει, αν υπάρχει ένα τουλάχιστον στοιχείο στη στοίβα, δηλαδή ελέγχει αν γίνεται *υποχείλιση* (underflow) της στοίβας. (σ.61)

9. Να δοθεί ο ορισμός της ουράς.

Ουρά ονομάζεται η δομή δεδομένων στην οποία τα δεδομένα που τοποθετήθηκαν πρώτα στην ουρά θα επεξεργαστούν πρώτα ενώ τα νέα δεδομένα που μπαίνουν στην ουρά τοποθετούνται στο τέλος της. Η μέθοδος αυτή επεξεργασίας ονομάζεται *Πρώτο μέσα, πρώτο έξω* (FIFO : First-In-First-Out). (σ.61)

10. Ποιες είναι οι βασικές λειτουργίες που γίνονται σε μία ουρά;

Οι κύριες λειτουργίες που εκτελούνται σε μία ουρά είναι :

- η εισαγωγή (enqueue) στοιχείου στο πίσω άκρο της ουράς, και
- η εξαγωγή (dequeue) στοιχείου από το εμπρός άκρο της ουράς. (σ.61)

11. Πώς μπορεί να υλοποιηθεί μια ουρά και τι έλεγχοι πρέπει να γίνονται;

Μια στοίβα μπορεί να υλοποιηθεί πολύ εύκολα με τη βοήθεια ενός μονοδιάστατου πίνακα. Στην περίπτωση της ουράς απαιτούνται δύο δείκτες: ο εμπρός (front) και ο πίσω (rear) δείκτης, που μας δίνουν τη θέση του στοιχείου που σε πρώτη ευκαιρία θα εξαχθεί και τη θέση του στοιχείου που μόλις εισήλθε.

Πρέπει να ελέγχεται αν υπάρχει ελεύθερος χώρος στον πίνακα για την εισαγωγή και αν υπάρχει ένα τουλάχιστον στοιχείο για την εξαγωγή. (σ.62)

12. Να περιγραφεί η λειτουργία της αναζήτησης.

Στην *αναζήτηση* (searching) ενός στοιχείου σε πίνακα θέλουμε να βρούμε τη θέση του στοιχείου στον πίνακα δοθείσης μιας τιμής (key). (σ.63)

Η μέθοδος της αναζήτησης που θα χρησιμοποιηθεί εξαρτάται κυρίως από το αν ο πίνακας είναι ταξινομημένος ή όχι. Μια άλλη παράμετρος είναι αν ο πίνακας περιέχει στοιχεία που είναι όλα διάφορα μεταξύ τους ή όχι.

13. Να δοθεί ένα παράδειγμα για τη σειριακή αναζήτηση στοιχείου σε έναν πίνακα. (σ64)

14. Πότε χρησιμοποιείται η σειριακή αναζήτηση;

Η σειριακή μέθοδος αναζήτησης είναι η πιο απλή, αλλά και η λιγότερη αποτελεσματική μέθοδος αναζήτησης. Έτσι, δικαιολογείται η χρήση της μόνο σε περιπτώσεις όπου:

- ο πίνακας είναι μη ταξινομημένος,
- ο πίνακας είναι μικρού μεγέθους (για παράδειγμα,  $n \leq 20$ ),
- η αναζήτηση σε ένα συγκεκριμένο πίνακα γίνεται σπάνια (σ.64)

15. Να δοθεί ο ορισμός της έννοιας της ταξινόμησης.

Δοθέντων των στοιχείων  $a_1, a_2, \dots, a_n$  η ταξινόμηση συνίσταται στη *μετάθεση* (permutation) της θέσης των στοιχείων, ώστε να τοποθετηθούν σε μία σειρά  $a_{k1}, a_{k2}, \dots, a_{kn}$  έτσι ώστε, δοθείσης μιας *συνάρτησης διάταξης* (ordering function),  $f$ , να ισχύει:  $f(a_{k1}) \leq f(a_{k2}) \leq \dots \leq f(a_{kn})$  (σ.65)

16. Να περιγραφεί η ταξινόμηση ευθείας ανταλλαγής και να δοθεί ένα παράδειγμα. (σ66,67)

17. Περιγράψτε τις δομές δεδομένων σε δευτερεύουσα μνήμη.

Τα στοιχεία ενός αρχείου ονομάζονται *εγγραφές* (records), όπου κάθε εγγραφή αποτελείται από ένα ή περισσότερα *πεδία* (fields). Ένα σύνολο από αρχεία αποτελούν μια βάση δεδομένων. (σ.66)

18. Τι γνωρίζεται για τα κλειδιά στα αρχεία;

Τα πεδία χωρίζονται σε αυτά που ταυτοποιούν την εγγραφή, και σε άλλα που περιγράφουν διάφορα χαρακτηριστικά της εγγραφής.

Το πεδίο που ταυτοποιεί την εγγραφή ονομάζεται *πρωτεύον κλειδί* (primary key) ή απλά κλειδί.

Αν υπάρχει πρωτεύον κλειδί και υπάρχει και άλλο πεδίο που ταυτοποιεί την εγγραφή, αυτό αποκαλείται *δευτερεύον κλειδί* (secondary keys). (σ.66)

19. Ποιες οι διαφορές στις δομές δεδομένων στην κύρια και δευτερεύουσα μνήμη;

Στη δευτερεύουσα μνήμη (δίσκους κλπ) αποθηκεύονται τα δεδομένα έχουμε μεγάλο όγκο δεδομένων και το μέγεθος της κύριας μνήμης δεν επαρκεί για την αποθήκευσή τους.

Επίσης τα δεδομένα δεν χάνονται αν διακοπεί η ηλεκτρική παροχή και διατηρούνται ακόμη και μετά τον τερματισμό ενός προγράμματος, κάτι που δεν συμβαίνει στην περίπτωση των δομών της κύριας μνήμης, όπως είναι οι πίνακες, όπου τα δεδομένα χάνονται όταν τελειώσει το πρόγραμμα. (σ.66)

20. Ποια η δομή του κόμβου μιας λίστας;

Οι κόμβοι μιας λίστας συνήθως βρίσκονται σε απομακρυσμένες θέσεις μνήμης και η σύνδεσή τους γίνεται με δείκτες. Ο δείκτης είναι ένα πεδίο κάθε κόμβου της δομής και οι τιμές του είναι διευθύνσεις στην κύρια μνήμη. Το πεδίο Δεδομένα μπορεί να περιέχει μία ή περισσότερες αλφαριθμητικές ή αριθμητικές πληροφορίες. (σ.71)

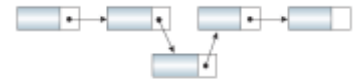
Δεδομένα	Δείκτης
----------	---------

Δομή κόμβου λίστας

21. Ποιες ενέργειες γίνονται στην εισαγωγή ενός κόμβου σε μια λίστα;  
Οι απαιτούμενες ενέργειες για την εισαγωγή (παρεμβολή) του νέου κόμβου είναι:

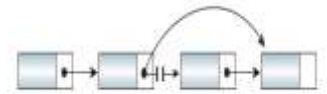
- ο δείκτης του δεύτερου κόμβου να δείχνει το νέο κόμβο
- και ο δείκτης του νέου κόμβου να δείχνει τον τρίτο κόμβο (δηλαδή να πάρει την τιμή που είχε πριν την εισαγωγή ο δείκτης του δεύτερου κόμβου).

Έτσι οι κόμβοι της λίστας διατηρούν τη λογική τους σειρά, αλλά οι φυσικές θέσεις στη μνήμη μπορεί να είναι τελείως διαφορετικές. (σ.72)



22. Ποιες ενέργειες γίνονται στην διαγραφή ενός κόμβου από μια λίστα;

Για τη **διαγραφή ενός κόμβου** αρκεί ν' αλλάξει τιμή ο δείκτης του προηγούμενου κόμβου και να δείχνει πλέον τον επόμενο αυτού που διαγράφεται, Ο κόμβος που διαγράφηκε αποτελεί "άχρηστο δεδομένο" και ο χώρος μνήμης που καταλάμβανε, παραχωρείται για άλλη χρήση. (σ.72)



23. Ποια τα κύρια χαρακτηριστικά δέντρων ως δομών;

Το κύριο χαρακτηριστικό των δένδρων είναι, ότι από έναν κόμβο δεν υπάρχει ένας μόνο επόμενος κόμβος, αλλά περισσότεροι. Υπάρχει ένας μόνο κόμβος, που λέγεται **ρίζα**, από τον οποίο ξεκινούν όλοι οι άλλοι κόμβοι οι οποίοι λέγονται παιδιά κ.ο.κ. (σ.72)



24. Τι είναι γράφοι;

Ένας **γράφος** (graph) αποτελείται από ένα **σύνολο κόμβων** (ή σημείων ή κορυφών) και ένα **σύνολο γραμμών** (ή ακμών ή τόξων) που ενώνουν μερικούς ή όλους τους κόμβους. (σ.73)

