

## Κεφάλαιο 4

# Τεχνικές Σχεδίασης Αλγορίθμων

### 4.1 Γενικός διδακτικός σκοπός

Ο γενικός σκοπός του κεφαλαίου είναι να κατανοήσουν οι μαθητές τις σύγχρονες τεχνικές σχεδίασης αλγορίθμων και να γίνουν ικανοί να αναγνωρίζουν την τεχνική που είναι κατάλληλη για την επίλυση προβλημάτων. Ο διδακτικός στόχος είναι να εξοικειωθούν οι μαθητές με τις διαφορετικές τεχνικές και να μπορούν να εντάξουν προβλήματα σε αλγορίθμους των τεχνικών αυτών, ώστε να βελτιώσουν την αποδοτικότητα επίλυσης ενός προβλήματος.

## 4.2 Ειδικοί διδακτικοί σκοποί

Μετά την ολοκλήρωση του παρόντος κεφαλαίου, οι μαθητές θα πρέπει να είναι σε θέση:

- να κατανοήσουν τις τεχνικές Διαίρει και Βασίλευε, Δυναμικό Προγραμματισμό και Άπληστη προσέγγιση.
- ανάπτυξη αλγορίθμων και παράθεση παραδειγμάτων για την επίλυση προβλημάτων μέσω των προσεγγίσεων αυτών.
- να αναγνωρίζουν και να διακρίνουν τα κυριότερα είδη των αλγοριθμικών τεχνικών που χρησιμοποιούνται για την επίλυση προβλημάτων
- να εντάσσουν ένα πρόβλημα στην κατάλληλη τεχνική επίλυσης και να μπορούν να τεκμηριώσουν την επιλογή της συγκεκριμένης τεχνικής.
- να χρησιμοποιούν την κατάλληλη τεχνική με στόχο τη βελτίωση της επίδοσης του προτεινόμενου αλγορίθμου.

## 4.3 Οδηγίες – επισημάνσεις

Ιδιαίτερη έμφαση και προσοχή πρέπει να δοθεί στις δύο προσεγγίσεις των αλγορίθμων: “από επάνω προς τα κάτω” και “από κάτω προς τα επάνω”. Οι δύο αυτές προσεγγίσεις αντιστοιχούν στους αλγορίθμους των οικογενειών “Διαίρει και Βασίλευε” και “Δυναμικού Προγραμματισμού”. Η πρώτη διάκριση εξυπηρετεί περισσότερο σκοπούς διδακτικής, ενώ η δεύτερη διάκριση είναι περισσότερο τεχνική και έτσι συναντάται στη βιβλιογραφία.

Είναι απαραίτητο να γίνουν αρκετά παραδείγματα για τις τεχνικές αυτές και να δοθεί έμφαση στον τρόπο επιλογής κάθε τεχνικής αναφέροντας τον τρόπο προσαρμογής της σε ένα δεδομένο πρόβλημα. Επίσης, είναι χρήσιμο να γίνει και πάλι συζήτηση για το πρόβλημα της ταξινόμησης και τους αντίστοιχους αλγορίθμους που δόθηκαν προηγουμένως. Για παράδειγμα, θα μπορούσε να θεωρηθεί ότι ο αλγόριθμος της ευθείας εισαγωγής (δες δραστηριότητα ΔΣ3 του προηγούμενου κεφαλαίου) εφαρμόζει την “άπληστη μέθοδο” με την έννοια ότι κάθε φορά εξετάζει τις διαθέσιμες επιλογές και βρίσκει κάθε φορά τη βέλτιστη. Ακόμη, θα μπορούσαν να δοθούν παραδείγματα που αφορούν σε αποτύπωση προβλημάτων σε μορφή γράφων και να τονισθεί η εφαρμογή τους σε ένα εύρος προβλημάτων.

## 4.4 Προγραμματισμός μαθημάτων κεφαλαίου

### Προτεινόμενος αριθμός μαθημάτων

δύο (2) δίωρα μαθήματα

### Σχέδιο 1ου μαθήματος

#### Διδακτικοί στόχοι

Διδακτικοί στόχοι του μαθήματος είναι οι μαθητές να μπορούν:

- να κατανοήσουν την τεχνική Διαίρει και Βασίλευε
- να γίνει ανάπτυξη αλγορίθμων και παράθεση παραδειγμάτων για την επίλυση προβλημάτων μέσω αυτής της προσέγγισης.
- να χρησιμοποιούν την κατάλληλη τεχνική με στόχο τη βελτίωση της επίδοσης του προτεινόμενου αλγορίθμου.

#### Χώρος υλοποίησης μαθήματος

τάξη

#### Προτεινόμενα υλικά και εποπτικά μέσα διδασκαλίας

πίνακας, διαφάνειες.

#### Περιεχόμενα θεωρητικής παρουσίασης

- **Ανάλυση προβλημάτων.** Ανάλυση προβλήματος με πρόταση για συγκεκριμένη μεθοδολογία και ακολουθία βημάτων. Πρόταση έξυπνων και αποδοτικών λύσεων.
- **Μέθοδοι σχεδίασης αλγορίθμων.** Καταγραφή των κυριοτέρων μεθόδων σχεδίασης αλγορίθμων. Εξήγηση της αναγκαιότητας ύπαρξης διαφορετικών τεχνικών προσέγγισης για την επίλυση προβλημάτων.

- **Μέθοδος διαιρεί και βασίλευε.** Περιγραφή της μεθόδου και εξήγηση της σπουδαιότητάς της και της κατηγορίας προβλημάτων που αφορά.
- **Δυαδική αναζήτηση.** Καταγραφή του αλγορίθμου της δυαδικής αναζήτησης και σχολιασμός της αποδοτικότητας του.

### **Περιεχόμενα πρακτικής εφαρμογής**

- **Εφαρμογές, παραδείγματα από το βιβλίο του μαθητή.** Θα πρέπει να διδαχθούν οι παράγραφοι 4.1 έως και 4.3 από το βιβλίο του μαθητή. Να απαντηθούν οι ερωτήσεις 1-4 του αντίστοιχου κεφαλαίου από το βιβλίο του μαθητή.
- **Δραστηριότητες από το τετράδιο του μαθητή.** Θα πρέπει να διδαχθούν τα παραδείγματα 1 και 2 από το τετράδιο του μαθητή.
- Κατ' ελάχιστον θα πρέπει να δοθούν στους μαθητές προς λύση μία από τις δραστηριότητες ΔΤ2 ή ΔΤ3 στην τάξη, και μία από τις δραστηριότητες ΔΣ1, ΔΣ2 ή ΔΣ4 για το σπίτι.

### **Τεστ αξιολόγησης επίδοσης**

#### **Συμπληρώστε με σωστό ή λάθος**

1. Η λύση σε ένα πρόβλημα μπορεί να προέλθει από ποικίλες και διαφορετικές προσεγγίσεις, τεχνικές και μεθόδους.
2. Η ανάλυση ενός προβλήματος σε ένα σύγχρονο υπολογιστικό περιβάλλον περιλαμβάνει την εισαγωγή και εξαγωγή δεδομένων.
3. Δεν υπάρχουν πολλά πρακτικά προβλήματα που να μπορούν να περιγραφούν με τη βοήθεια ενός διαγράμματος ή γράφου.
4. Ο γράφος αποθηκεύεται (συνήθως) στη μνήμη του υπολογιστή με τη βοήθεια ενός δισδιάστατου πίνακα.
5. Υπάρχει ένας ενιαίος κανόνας που να αναφέρεται στην επίλυση του συνόλου των προβλημάτων.
6. Στην κατηγορία “Διαιρεί και Βασίλευε” εντάσσονται οι τεχνικές που αναλύουν ένα πρόβλημα σε μεγαλύτερα προβλήματα.

7. Ο αλγόριθμος της Δυαδικής αναζήτησης εφαρμόζεται σε έναν μη ταξινομημένο πίνακα.
8. Με την τεχνική “Διαίρει και Βασίλευε” υποδιαιρείται το στιγμιότυπο του προβλήματος σε υπο-στιγμιότυπα του ίδιου προβλήματος.

**Επιλέξτε μεταξύ των προτεινόμενων μία σωστή απάντηση**

9. Η ανάλυση προβλημάτων περιλαμβάνει:
  - α) καταγραφή υπάρχουσας πληροφορίας
  - β) καταγραφή αποτελεσμάτων
  - γ) αναγνώριση στοιχείων εισόδου του προβλήματος
  - δ) πρόταση για την είσοδο και έξοδο των δεδομένων
10. Ένας από τους αλγορίθμους που εντάσσονται στην κατηγορία «Διαίρει και Βασίλευε» είναι:
  - α) ταξινόμηση με ευθεία ανταλλαγή
  - β) ταξινόμηση με ευθεία επιλογή
  - γ) δυαδική αναζήτηση
  - δ) υπολογισμός δύναμης
11. Κάθε τεχνική αλγορίθμων πρέπει να έχει:
  - α) τη δική της υπολογιστική μηχανή
  - β) τη δική της ακολουθία εντολών
  - γ) τη δική της είσοδο και έξοδο
  - δ) τη δική της γλώσσα προγραμματισμού

**Απαντήσεις τεστ αξιολόγησης επίδοσης**

1: σωστό	5: λάθος	9: α
2: λάθος	6: λάθος	10: γ
3: λάθος	7: λάθος	11: β
4: σωστό	8: σωστό	

## Σχέδιο 2ου μαθήματος

### Διδακτικοί στόχοι

Διδακτικοί στόχοι του μαθήματος είναι οι μαθητές να μπορούν:

- να κατανοήσουν τις τεχνικές του Δυναμικού Προγραμματισμού και της Άπληστης προσέγγισης
- ανάπτυξη αλγορίθμων και παράθεση παραδειγμάτων για την επίλυση προβλημάτων μέσω των δύο αυτών προσεγγίσεων.
- να αναγνωρίζουν και να διακρίνουν όλα τα είδη των αλγοριθμικών τεχνικών που χρησιμοποιούνται για την επίλυση προβλημάτων
- να εντάσσουν ένα πρόβλημα στην κατάλληλη τεχνική επίλυσης και να μπορούν να τεκμηριώσουν την επιλογή της συγκεκριμένης τεχνικής.
- να χρησιμοποιούν την κατάλληλη τεχνική με στόχο τη βελτίωση της επίδοσης του προτεινόμενου αλγορίθμου.

### Χώρος υλοποίησης μαθήματος

τάξη

### Προτεινόμενα υλικά και εποπτικά μέσα διδασκαλίας

πίνακας, διαφάνειες.

### Περιεχόμενα θεωρητικής παρουσίασης

- **Δυναμικός προγραμματισμός.** Εισαγωγή στη λογική και στα βήματα που ακολουθεί η τεχνική του Δυναμικού Προγραμματισμού.
- **Άπληστη Μέθοδος.** Επεξηγείται και δίνεται έμφαση στην αναγκαιότητα της μεθόδου αυτής για την επίλυση προβλημάτων βελτιστοποίησης.
- **Επανάληψη των εννοιών που διδάχθηκαν – Ανακεφαλαίωση.**

**Περιεχόμενα πρακτικής εφαρμογής**

- **Εφαρμογές, παραδείγματα από το βιβλίο του μαθητή.** Θα πρέπει να διδαχθούν οι παράγραφοι 4.4 έως και 4.5 από το βιβλίο του μαθητή. Να απαντηθούν οι ερωτήσεις 5-8 από το βιβλίο του μαθητή.
- **Δραστηριότητες από το τετράδιο του μαθητή.** Θα πρέπει να διδαχθεί το παράδειγμα 3 από το τετράδιο του μαθητή.
- Κατ' ελάχιστον θα πρέπει να δοθούν στους μαθητές προς λύση οι δραστηριότητες ΔΤ1 στην τάξη και μία-δύο από τις δραστηριότητες ΔΣ3, ΔΣ5 και ΔΣ6 για το σπίτι.

**Τεστ αξιολόγησης επίδοσης****Συμπληρώστε με σωστό ή λάθος**

1. Η φιλοσοφία της τεχνικής του Δυναμικού προγραμματισμού από την αρχή να επιλύονται τα μεγαλύτερα προβλήματα και σταδιακά να επιλύονται τα μικρότερα.
2. Η μέθοδος του Δυναμικού προγραμματισμού χρησιμοποιείται κυρίως για την επίλυση προβλημάτων βελτιστοποίησης, δηλαδή χρησιμοποιείται κυρίως όταν χρειάζεται να βρεθεί το ελάχιστο ή το μέγιστο κάποιου μεγέθους.
3. Η μέθοδος της Άπληστης προσέγγισης προχωρά με βάση σταδιακές επιλογές που αφορούν το βέλτιστο κάθε βήματος, χωρίς μέριμνα για το τελικό βέλτιστο.

**Συμπλήρωσε τα κενά με το σωστή λέξη που λείπει**

4. Η φιλοσοφία της τεχνικής \_\_\_\_\_ είναι από την αρχή να επιλύονται τα μικρότερα προβλήματα και σταδιακά να επιλύονται τα μεγαλύτερα ως σύνθεση των απλούστερων.
5. Η μέθοδος της Άπληστης προσέγγισης προχωρά με βάση σταδιακές επιλογές που αφορούν το \_\_\_\_\_ κάθε βήματος, χωρίς μέριμνα για το τελικό \_\_\_\_\_.
6. Η \_\_\_\_\_ μέθοδος, μπορεί να τυποποιηθεί σύμφωνα με την παραδοχή ότι σε κάθε βήμα γίνεται επιλογή της τρέχουσας βέλτιστης επιλογής και υ-

πάρχει επιβεβαίωση ότι αυτή η προσέγγιση εγγυάται τη συνολική βέλτιστη λύση.

#### **Απαντήσεις τεστ αξιολόγησης επίδοσης**

1: λάθος	4: Δυναμικού προγραμματισμού
2: σωστό	5: βέλτιστο
3: σωστό	6: άπληστη

### **4.5 Προβληματισμοί και θέματα προς συζήτηση**

- Είναι χρήσιμο να τονισθεί η ιδιαίτερη χρησιμότητα των τεχνικών σχεδίασης των αλγορίθμων και να ζητηθεί από τους μαθητές να αναφερθούν σε παραδείγματα από την καθημερινή τους ζωή, όπου θα μπορούσαν να εντάξουν σε κατάλληλη τεχνική επίλυσης.
- Να συζητηθεί το πρόβλημα της ταξινόμησης με έμφαση στην ύπαρξη διαφορετικών αλγορίθμων και στις διαφορές και ιδιαιτερότητές τους.

### **4.6 Προτεινόμενες πηγές πληροφόρησης**

Όλη η προτεινόμενη βιβλιογραφία του κεφαλαίου, όπως καταγράφεται στο βιβλίο του μαθητή.

### **4.7 Απαντήσεις ερωτήσεων βιβλίου μαθητή**

- 1: Δες παράγραφο 4.1 του βιβλίου (αρχή παραγράφου).
- 2: Δες παράγραφο 4.1 του βιβλίου (τελευταία παράγραφος)
- 3: Παρατηρώντας τον τρόπο με τον οποίο οι διάφορες τεχνικές υλοποιούνται παρατηρούμε ότι υπάρχουν δύο προσεγγίσεις: η «από πάνω προς τα κάτω» (ανάλυση σε μικρότερα υποπροβλήματα) και η «από κάτω προς τα επάνω» (σύνθεση από μικρότερα υποπροβλήματα).



- 4: Δες παράγραφο 4.3 του βιβλίου.
- 5: Δες παράγραφο 4.4 του βιβλίου.
- 6: Δες παράγραφο 4.3 του βιβλίου.
- 7: Πρέπει να θεωρηθεί ένας πίνακας  $N$  θέσεων αν πρόκειται να υπολογισθεί το  $N!$ . Σε κάθε θέση του πίνακα τοποθετούμε το παραγοντικό της αντίστοιχης τιμής. Να συζητηθεί ότι αυτή η μέθοδος δεν μπορεί να εφαρμοσθεί για μεγάλο  $N$ , λόγω του μεγέθους του ακεραίου (τέσσερις χαρακτήρες).
- 8: Ο υπολογισμός της ακέραιας δύναμης ενός πραγματικού αριθμού,  $a^b$ , όπου το  $a$  είναι ακέραιος και  $b$  πραγματικός παρουσιάστηκε στο Κεφάλαιο 3. Ακολουθεί ένας νέος απλός (ίσως απλοϊκός) τρόπος με χρήση της φιλοσοφίας “από κάτω προς τα επάνω” (δυναμικός προγραμματισμός), όπου η βασική ιδέα είναι η αποθήκευση των προηγούμενων δυνάμεων του αριθμού (1, 2, ...,  $b-1$ ) μέχρι την “από κάτω προς τα επάνω” κατάληξη στη δύναμη  $b$ .

**Αλγόριθμος** Δύναμη

**Δεδομένα** \\  $a, b$  \\

$power[1] \leftarrow 1$

**Για**  $i$  **από** 1 **μέχρι**  $b$

$power[i] \leftarrow power[i] * a$

**Τέλος\_επανάληψης**

**Αποτελέσματα** \\  $power[b]$  \\

**Τέλος** Δύναμη3

Ο αλγόριθμος αυτός μπορεί να βελτιωθεί με χρήση ενδιάμεσων αποτελεσμάτων δυνάμεων χωρίς να αποθηκεύονται όλες οι δυνάμεις. Συγκεκριμένα ο πίνακας  $power$  θα αποθηκεύει το αποτέλεσμα της ύψωσης στο τετράγωνο της προηγούμενης θέσης του πίνακα. Για παράδειγμα, αν θα υπολογιζόταν οι δυνάμεις του 2, τότε ο πίνακας θα είχε την τιμή 1 στη θέση 0, την τιμή 2 στη θέση 1, την τιμή 4 στη θέση 2, την τιμή 16 στη θέση 3 κ.ο.κ. όπως φαίνεται και στο επόμενο σχήμα.

	0	1	2	3	4	...
	1	2	4	16	256	...
Δύναμη του 2	0η	1η	2η	4η	8η	...

Για τον υπολογισμό κάποιας δύναμης απαιτούνται οι κατάλληλες “από κάτω” θέσεις του πίνακα. Για την ανεύρεση του  $2^7$ , αρκούν οι αποθηκευμένες δυνάμεις

μέχρι και την 4η δύναμη του 2, διότι  $2^7 = 2^4 * 2^2 * 2^1$ . Ο αλγόριθμος που ακολουθεί παρουσιάζει τον υπολογισμό του  $a^b$  με τη νέα αυτή ευφυέστερη τεχνική που βασίζεται στο δυναμικό προγραμματισμό.

```

Αλγόριθμος Δύναμη
Δεδομένα \\ a, b \\
power[1] ← a
i ← 1
pow ← 1
Οσο power < b επανάλαβε
    i ← i+1
    pow ← 2*pow
    power[i] ← power[i-1]*power[i-1]
Τέλος_επανάληψης
used ← 0
result ← 1
Οσο used < b επανάλαβε
    Αν used+pow <= n τότε
        result ← result*power[i]
        used ← used+pow
        pow ← pow/2
    Τέλος_αν
    i ← i-1
Τέλος_επανάληψης
Αποτελέσματα \\ result \\
Τέλος Δύναμη

```

## 4.8 Απαντήσεις δραστηριοτήτων τετραδίου μαθητή

### ➤ Στην τάξη

#### ΔΤ1.

Μπορεί να προταθεί βελτίωση με μετακίνηση της ελάχιστης τιμής στην πρώτη θέση μετά από ένα πρώτο κύκλο εκτέλεσης. Αυτή η τοποθέτηση της ελάχιστης τιμής στην πρώτη θέση του πίνακα προϋποθέτει την ανταλλαγή του στοιχείου που υπήρχε στην πρώτη θέση με τη θέση του πίνακα που είχε την ελάχιστη τιμή.

Επομένως ο δεύτερος κύκλος εκτέλεσης ξεκινά από τη δεύτερη θέση του πίνακα. Ο αλγόριθμος Ανταλλαγή( $M[1], M[pos]$ ) πραγματοποιεί την ανταλλαγή των θέσεων  $M[1]$  και  $M[pos]$ :

```
Αλγόριθμος Δύο_Μικρότεροι
Δεδομένα // M //
low1 ← M[1]
pos ← 1
Για i από 2 μέχρι 50
    Αν M[i] < low1 τότε
        low1 ← M[i]
        pos ← i
    Τέλος_αν
Τέλος_επανάληψης
Αντιμετάθεση(M[1], M[pos])
low2 ← M[2]
Για i από 3 μέχρι 50
    Αν M[i] < low2 τότε low2 ← M[i]
Τέλος_επανάληψης
Αποτελέσματα // low1 , low2 //
Τέλος Δύο_Μικρότεροι
```

#### Δ72.

Εδώ μπορεί να χρησιμοποιηθεί ο αλγόριθμος της σειριακής αναζήτησης που παρουσιάστηκε στην Παράγραφο 3.6 του βιβλίου και ο αλγόριθμος της Δυαδικής αναζήτησης που παρουσιάστηκε στην Παράγραφο 4.3 του βιβλίου και δόθηκε αναλυτικά στο Παράδειγμα 2 του Κεφαλαίου 4 του τετραδίου του μαθητή. Η δυαδική αναζήτηση προϋποθέτει την ταξινόμηση του πίνακα με τα μουσεία της πόλης.

#### Δ73.

Χρειάζεται να παρακολουθήσετε στην τάξη τον αλγόριθμο που δίνεται και να σχολιάσετε την πορεία του. Ο αλγόριθμος εντάσσεται στην κατηγορία της άπληστης μεθόδου αφού σε κάθε βήμα γίνεται επιλογή της βέλτιστης θέσης με στόχο τη συνολική βέλτιστη λύση.

**ΔΤ4.**

Η άσκηση αυτή αντιστοιχεί στο γνωστό πρόβλημα του περιοδεύοντος πωλητή. Όπως είναι γνωστό, το πρόβλημα αυτό γενικά είναι δυσχεύριστο (εκτός ειδικών περιπτώσεων). Στο σημείο αυτό μπορεί να γίνει μία ευριστική επίλυση με γραφικό τρόπο στον πίνακα υιοθετώντας μία άπληστη τεχνική.

**► Στο σπίτι****ΔΣ1.**

Το πρόβλημα αυτό μπορεί να επιλυθεί με τη μέθοδο Διαίρει και Βασίλευε ξεκινώντας με άνω όριο το 10 και κάτω όριο το 0 αφού είναι γνωστά τα όρια του αριθμού που θα δοθεί.

```
Αλγόριθμος Μαντεύω_Αριθμό
Δεδομένα // number //
low ← 0
high ← 10
found ← 0
Διάβασε number
Όσο low <= high επανάλαβε
    mid ← (low + high)/2
    Αν mid < number τότε
        low ← mid+1
    αλλιώς_αν mid > number τότε
        high ← mid-1
    αλλιώς
        found ← 1
    Τέλος_αν
Τέλος_επανάληψης
Αποτελέσματα // found //
Τέλος Μαντεύω_Αριθμό
```

**ΔΣ2.**

Χρειάζεται να ακολουθηθεί ο αλγόριθμος της δυαδικής αναζήτησης που παρουσιάστηκε στην Παράγραφο 4.3 του βιβλίου και δόθηκε αναλυτικά στο Παράδειγμα 2 του Κεφαλαίου 4 του τετραδίου του μαθητή.

**ΔΣ3.**

Ο αλγόριθμος υπολογίζει τη δύναμη ακεραίου σε ακέραιο  $x^n$

**ΔΣ4.**

Χρειάζεται να ακολουθηθεί ο αλγόριθμος της δυαδικής αναζήτησης που παρουσιάστηκε στην Παράγραφο 4.3 του βιβλίου και δόθηκε αναλυτικά στο Παράδειγμα 2 του Κεφαλαίου 4 του τετραδίου του μαθητή.

**ΔΣ5.**

Χρειάζεται να ταξινομηθούν οι βαθμοί κάθε χώρας με χρήση κάποιου από τους αλγορίθμους ταξινόμησης που είναι ήδη γνωστοί. Εστω ότι ο πίνακας VATH έχει ταξινομηθεί. Θεωρούμε ότι συνολικά στο διαγωνισμό παίρνουν μέρος 20 Ευρωπαϊκές χώρες. Στη συνέχεια πρέπει να υπάρχει το ακόλουθο τμήμα του αλγορίθμου που θα επιλέγει τις χώρες ώστε το άθροισμα της βαθμολογίας όλων των τραγουδιών που θα προχωρήσουν στη δεύτερη φάση να είναι μικρότερο από 1000 βαθμούς:

```
SUM ← VATH[20]
Για i από 19 μέχρι 1 με βήμα -1
    Αν SUM+VATH[i] < 1000 τότε
        SUM ← SUM+VATH[i]
    αλλιώς
        stop ← i-1
        έξοδος από βρόχο
Τέλος_αν
Τέλος_επανάληψης
```

Η μεταβλητή stop θα δείχνει τη θέση από το τέλος του πίνακα που θα εκφράζει την τελευταία χώρα σε βαθμολογία που θα συμμετάσχει στο διαγωνισμό αφού ο πίνακας είναι ταξινομημένος σε αύξουσα τάξη.

#### ΔΣ6.

Εξηγήθηκε στην ΔΤ5 παραπάνω.

## 4.9 Συμπληρωματικά στοιχεία

Δίνουμε εδώ μερικά επιπλέον στοιχεία από την ύλη του κεφαλαίου.

### 4.9.1 Γρήγορη ταξινόμηση

Αρκετές μέθοδοι ταξινόμησης βασίζονται στη φιλοσοφία διαίρει και βασιλεύε. Μία μέθοδος ταξινόμησης αυτού του τύπου είναι ιδιαίτερα αποτελεσματική και για το λόγο αυτό ονομάστηκε **γρήγορη ταξινόμηση** (quicksort). Η γρήγορη ταξινόμηση στηρίζεται στην παρατήρηση ότι είναι προτιμότερο οι ανταλλαγές να γίνονται μεταξύ απομακρυσμένων στοιχείων ενός πίνακα, παρά μεταξύ στοιχείων γειτονικών θέσεων, όπως συμβαίνει στην περίπτωση των μεθόδων ταξινόμησης με ευθεία ανταλλαγή (bubblesort) και με ευθεία εισαγωγή. Ας υποθεθεί ότι ο πίνακας αποτελείται από  $n$  στοιχεία που είναι ταξινομημένα κατά φθίνουσα τάξη της τιμής των κλειδιών τους. Τα στοιχεία αυτά είναι δυνατόν να ταξινομηθούν με  $n/2$  ανταλλαγές μόνο, αντιμετωπίζοντας αρχικά το πιο αριστερό με το πιο δεξιό στοιχείο και προχωρώντας σταδιακά και από τα δύο άκρα του πίνακα προς τα μεσαία στοιχεία. Φυσικά αυτό μπορεί να γίνει όταν είναι εκ των προτέρων γνωστό ότι τα στοιχεία είναι ταξινομημένα σε φθίνουσα τάξη. Ωστόσο το παράδειγμα αυτό μας διδάσκει κάτι χρήσιμο.

Στη συνέχεια δίνεται ο αλγόριθμος Γρήγορη\_Ταξινόμηση. Η βασική ιδέα του αλγορίθμου αυτού είναι η λήψη του πρώτου στοιχείου του πίνακα και η μετακίνησή του στη θέση, όπου τελικά θα αποθηκευθεί ανάλογα με την τιμή του κλειδιού του. Ταυτόχρονα με την αποθήκευση του στοιχείου αυτού στην τελική του θέση γίνεται αναδιάταξη των υπόλοιπων στοιχείων, έτσι ώστε να μην υπάρχει κανένα μικρότερο κλειδί προς τα δεξιά του και κανένα μεγαλύτερο κλειδί προς τα αριστερά του. Έτσι, το πρώτο αυτό στοιχείο παίζει το ρόλο του άξονα και ο πίνακας έ-

χει διαμερισθεί κατά τέτοιο τρόπο, ώστε το αρχικό πρόβλημα έχει να αναχθεί σε δύο απλούστερα προβλήματα, δηλαδή στην ανεξάρτητη ταξινόμηση των δύο υποπινάκων. Η απόσταση μεταξύ του διαμερισμού αυτού και της τελικής ταξινόμησης είναι πολύ μικρή. Μετά το διαμερισμό του πίνακα, ο ίδιος αλγόριθμος εφαρμόζεται στους δύο υποπίνακες, έπειτα στους υποπίνακες των υποπινάκων, κοκ., μέχρις ότου ο κάθε υποπίνακας να αποτελείται από ένα μόνο στοιχείο. Προφανώς, στο σημείο αυτό ο αρχικός πίνακας έχει ταξινομηθεί.

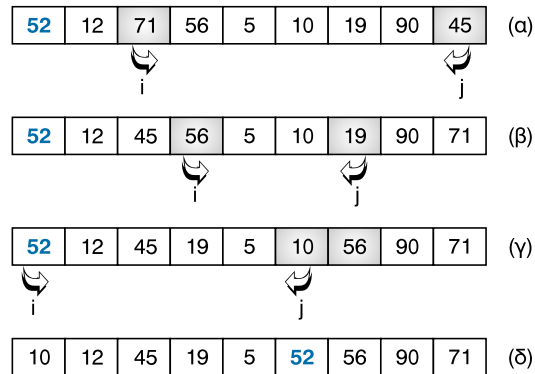
```

Αλγόριθμος Γρήγορη_ταξινόμηση
Δεδομένα // table, left, right //
Αν right > left τότε
    i ← left
    j ← right+1
    pivot ← table[left]
    Αρχή_επανάληψης
        Αρχή_επανάληψης
            i ← i+1
        Μέχρις ότου table[i] ≥ pivot
        Αρχή_επανάληψης
            j ← j-1
        Μέχρις ότου table[j] ≤ pivot
        Αν i < j τότε Αντιμετάθεσε(table[i], table[j])
    Μέχρις ότου i ≥ j
    Αντιμετάθεσε(table[left], table[j])
    Γρήγορη_ταξινόμηση(table, left, j-1)
    Γρήγορη_ταξινόμηση(table, j+1, right)
Τέλος_Αν
Αποτελέσματα // table //
Τέλος Γρήγορη_ταξινόμηση

```

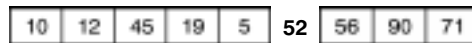
**Παράδειγμα.** Έστω ότι δίνεται για ταξινόμηση ο γνωστός από το προηγούμενο κεφάλαιο πίνακας με εννέα στοιχεία (Σχ. 4.1α). Παίρνουμε το πρώτο στοιχείο του πίνακα ως άξονα, δηλαδή ισχύει  $\text{pivot} = 52$  και συγκρίνουμε κάθε ένα από τα υπόλοιπα στοιχεία με το στοιχείο αυτό. Πιο συγκεκριμένα, ξεκινώντας από τα αριστερά αναζητούμε το πρώτο στοιχείο που είναι μεγαλύτερο από τον άξονα 52. Το στοιχείο αυτό είναι το 71 (δείκτης i). Κατόπιν, από τα δεξιά αναζητούμε το πρώτο στοιχείο που είναι μικρότερο από το 52, οπότε βρίσκουμε το 45 (δείκτης j). Στη συνέχεια εκτελούμε μία αντιμετάθεση των στοιχείων 71 και 45, οπότε προκύπτει η επόμενη μορφή του πίνακα, όπως φαίνεται στο σχήμα 4.1β. Η διαδικα-

σία συνεχίζεται σαρώνοντας από τα αριστερά για μεγαλύτερα του 52 στοιχεία και από τα δεξιά για μικρότερα του 52 στοιχεία. Έτσι, βρίσκουμε τα στοιχεία 56 (δείκτης  $i$ ) και 19 (δείκτης  $j$ ), τα οποία αντιμετατίθενται (Σχ. 4.1γ). Συνεχίζοντας τη σάρωση από τα αριστερά βρίσκουμε το στοιχείο 56 (δείκτης  $i$ ) και από τα δεξιά το στοιχείο 10 (δείκτης  $j$ ). Όμως στο σημείο αυτό ισχύει  $i > j$ , οπότε δεν εκτελείται αυτή η ανταλλαγή. Αντ'αυτής εκτελείται η αντιμετάθεση του στοιχείου του άξονα με το στοιχείο όπου σταμάτησε ο δείκτης  $j$ . Έτσι, ανταλλάσσονται τα στοιχεία 52 και 10, προκύπτει μία νέα μορφή της δομής που παρουσιάζεται στον επόμενο πίνακα (Σχ. 4.1δ).



**Σχήμα 4.1** Κατάσταση πίνακα κατά τις διαδοχικές συγκρίσεις και αντιμεταθέσεις κατά την πρώτη κλήση της διαδικασίας Γρήγορη\_ταξινόμηση.

Το αποτέλεσμα αυτών των συγκρίσεων είναι να υπάρχουν δύο υπο-πίνακες. Ο αριστερός υπο-πίνακας αποτελείται από πέντε στοιχεία, που είναι μικρότερα του 52. Το τελευταίο έχει τοποθετηθεί στην τελική του θέση στον πίνακα, και δεν πρόκειται να "ασχοληθούμε" στο μέλλον μαζί του. Στα δεξιά του 52 θεωρούμε το δεύτερο υπο-πίνακα, που αποτελείται από 3 στοιχεία με τιμές μεγαλύτερες του 52. Η κατάσταση αυτή παρουσιάζεται στο επόμενο σχήμα.



**Σχήμα 4.2** Κατάσταση πίνακα μετά την πρώτη κλήση της διαδικασίας Γρήγορη\_ταξινόμηση.



Στη συνέχεια καλείται αναδρομικά η διαδικασία Γρήγορη\_ταξινόμηση για τον αριστερό υποπίνακα και για το δεξιό υποπίνακα.

### 4.9.2 Οπισθοδρόμηση

Μια πολύ σημαντική τεχνική σχεδίασης αλγορίθμων είναι η οπισθοδρόμηση. Η τεχνική αυτή κάνει χρήση αναδρομής, συχνά σε δομές γράφων και γι' αυτό θεωρείται προωθημένο θέμα.

Η τεχνική της **οπισθοδρόμησης** (backtracking) στηρίζεται στην αρχή ότι, αναζητώντας τη λύση μέσα σε ένα σύνολο πιθανών περιπτώσεων, αν ανακαλυφθεί ότι συνεχίζοντας στην ίδια πορεία δεν είναι δυνατόν να βρεθεί η λύση, τότε επανέρχόμαστε σε ένα προηγούμενο σημείο και δοκιμάζουμε καινούριο δρόμο σε σχέση με την εξέταση των λύσεων. Η μέθοδος της οπισθοδρόμησης έχει εφαρμογές σε προβλήματα αναζήτησης σε γράφους, όπως για παράδειγμα στο γράφο του σχήματος 4.1 (βιβλίου μαθητή) που απεικονίζει τη σύνδεση πόλεων. Γενικά, η τεχνική αυτή μπορεί να συνοψισθεί στα εξής βήματα:

- 1) Ανίχνευση κάθε κόμβου του γράφου, με βάση κάποια συγκεκριμένη σειρά μέχρις ότου
  - ⇒ ή βρεθεί η βέλτιστη λύση (ή γενικώς η λύση),
  - ⇒ ή διαπιστωθεί ότι αποκλείεται η μέχρι τώρα πορεία, να δώσει τη λύση στη συνέχεια.
- 2) Στη δεύτερη περίπτωση, η διαδικασία επίλυσης “αναδιπλώνεται”, ώστε να προσεγγίσει άλλο κόμβο που να μπορεί να οδηγήσει στη λύση.

### Διάσχιση λαβυρίνθου

Ο λαβύρινθος είναι ένα σύνολο διαδρόμων που χωρίζονται από ψηλούς φράκτες, θάμνους ή τοίχους και συναντώνται σε διασταυρώσεις. Πρέπει, λοιπόν, εισερχόμενοι από τη μοναδική είσοδο του λαβυρίνθου να διατρέξουμε όλους τους διαδρόμους και να επιστρέψουμε στη μοναδική είσοδο/έξοδο. Για να μην εγκλωβισθούμε σε ένα λαβυρίνθο για πάντα (θεωρητικά), πρέπει να διατυπώσουμε ένα συστηματικό τρόπο διάσχισής του, δηλαδή έναν αλγόριθμο.

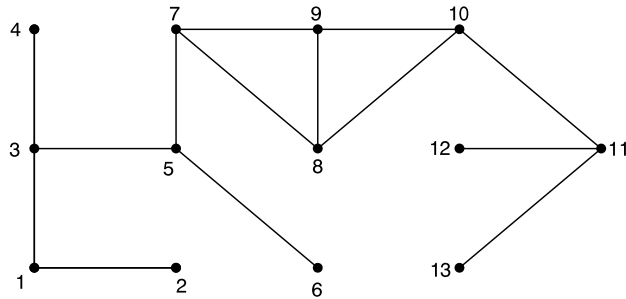
Το πρόβλημα του λαβυρίνθου έρχεται από το βάθος της ελληνικής μυθολογίας. Όπως είναι γνωστό, ο Ανδρόγειος, γιός του βασιλιά Μίνωα κέρδισε νίκες στα Παναθήναια, αλλά οι αντίπαλοί του τον σκότωσαν. Ο Μίνωας κήρυξε τον πόλεμο στην Αθήνα και την κυρίευσε. Οι Αθηναίοι αναγκάστηκαν κάθε χρόνο να στέλνουν 7 αγόρια και 7 κορίτσια θυσία στο Μινώταυρο που ήταν εγκλωβισμένος στο λαβύρινθο. Ο Θησέας, γιός του βασιλιά της Αθήνας Αιγαία, δέχθηκε να αντιμετωπίσει το Μινώταυρο. Με τη βοήθεια της Αριάδνης, κόρης του Μίνωα, κατάφερε να σκοτώσει το Μινώταυρο και να βγει από το λαβύρινθο χρησιμοποιώντας το μίτο.



**Σχήμα 4.3.** Ο κήπος του Hampton Court Palace

Κατά την Αναγέννηση πολλοί βασιλιάδες διακοσμούσαν τους κήπους των ανακτόρων τους με λαβυρίνθους. Ο πιο φημισμένος λαβύρινθος είναι αυτός του Hampton Court Palace, που βρίσκεται πολύ κοντά στο Λονδίνο και κατασκευάστηκε κατά παραγγελία του Γουλιέλμου του 3ου της Οράγγης. Ο λαβύρινθος αυτός παρουσιάζεται στο σχήμα 4.3. Το πρόβλημα της εύρεσης μεθόδου για την διάσχιση ενός λαβυρίνθου απασχόλησε πολλούς μαθηματικούς, αλλά λύθηκε μόνο στο τέλος του 19ου αιώνα.

Δεδομένου ενός λαβυρίνθου, προχωρούμε σε μία αφαιρετική αναπαράστασή του, δηλαδή κατασκευάζουμε ένα γράφο. Οι κορυφές του γράφου αντιστοιχούν στις διασταυρώσεις όπου συναντώνται περισσότεροι από δύο διάδρομοι, ενώ οι συνδέσεις των κορυφών αντιστοιχούν στους διαδρόμους του λαβυρίνθου. Στο σχήμα 4.4 παρουσιάζεται ο γράφος του αντιστοιχεί στο λαβύρινθο του σχήματος 4.3.



Σχήμα 4.4

Έχοντας, λοιπόν, τον αντίστοιχο γράφο πρέπει να διατυπώσουμε ένα αλγόριθμο συστηματικής επίσκεψης των κορυφών του. Ένας αλγόριθμος που επιτυγχάνει αυτό το στόχο λέγεται “αναζήτηση κατά βάθος” (depth-first search) και στηρίζεται στη φιλοσοφία της οπισθοδρόμησης. Όπως εξηγήθηκε στην εισαγωγή της παραγράφου, ακολουθούμε μία διαδρομή μέχρι να διαπιστώσουμε, ότι δεν είναι δυνατόν να συνεχίσουμε από εκείνο σημείο, οπότε επιστρέφουμε σε μία προηγούμενη κορυφή, για να συνεχίσουμε από εκεί προς μία νέα κατεύθυνση.

Ο επόμενος αναδρομικός αλγόριθμος υλοποιεί την αναζήτηση κατά βάθος και ονομάζεται ΑΚΒ, από τα αρχικά της μεθόδου. Ο αλγόριθμος αυτός θεωρεί ότι ο γράφος έχει  $n$  κορυφές και είναι αποθηκευμένος στη μνήμη του υπολογιστή με τη μέθοδο του πίνακα γειτνίασης. Επίσης, χρησιμοποιείται ένας βοηθητικός μονοδιάστατος πίνακας  $mark$ , που αποτελείται από  $n$  στοιχεία τύπου Αληθής/Ψευδής. Ο πίνακας αυτός χρησιμοποιείται για να σημειώνουμε, αν έχουμε επισκεφθεί κάποια κορυφή. Με απλά λόγια, κάθε στοιχείο του πίνακα, που προφανώς αντιστοιχεί σε κάποια κορυφή, αρχικά έχει την τιμή Ψευδής και λαμβάνει την τιμή Αληθής αν επισκεφθούμε την αντίστοιχη κορυφή.

**Αλγόριθμος** ΑΚΒ( $k$ )

**Δεδομένα** //  $a$  πίνακας γειτνίασης γράφου,  $n$  //

$mark(k) = \text{Αληθής}$

**Εμφάνισε**  $k$

**Για**  $j$  **από** 1 **μέχρι**  $n$

**Αν**  $a(k, j) = 1$  **και**  $mark(j) = \text{Ψευδής}$  **τότε** ΑΚΒ( $j$ )

**Τέλος\_επανάληψης**

**Αποτελέσματα** // Λίστα κόμβων γράφου //

**Τέλος** ΑΚΒ

Σε σχέση με το γράφο του σχήματος, η διαδικασία αρχίζει με την κλήση  $AKB(1)$ , δηλαδή από την κορυφή 1, που μαρκάρεται “ως επισκεφθείσα” θέτοντας  $mark(1)=\text{Αληθής}$ . Κατόπιν επιλέγεται η κορυφή 2 που είναι γειτονική της κορυφής 1 (επειδή ισχύει  $a(1,2)=1$ ) και καλείται αναδρομικά η διαδικασία  $AKB(2)$ . Έτσι η κορυφή 2 μαρκάρεται ως επισκεφθείσα (δηλαδή τίθεται  $mark(2)=\text{Αληθής}$ ), και η μέθοδος συνεχίζει από την κορυφή 2. Όμως από την κορυφή αυτή δεν υπάρχει άλλη γειτονική με μεγαλύτερη (του 2) επιγραφή, οπότε ο έλεγχος επιστρέφει (backtrack) στην κορυφή 1. Κατόπιν συνεχίζουμε με την κορυφή 3 καλώντας  $AKB(3)$ , επειδή ισχύει  $a(1,3)=1$  και από εκεί στην κορυφή 4 εκτελώντας  $AKB(4)$ . Όμως από την κορυφή 4 ο έλεγχος επιστρέφει (οπισθοδρομεί) στην κορυφή 3, οπότε εκτελείται η  $AKB(5)$ . Έτσι ο αλγόριθμος συνεχίζει με τη λογική αυτή, οπότε επισκεπτόμαστε τις κορυφές με τη σειρά 1, 2, 3, 4, 5, 6, 5, 7, 8, 9, 10, 11, 12, 13, που ταυτίζεται με τη σειρά απόδοσης της τιμής “Αληθής” στις αντίστοιχες θέσεις του πίνακα  $mark$ . Ωστόσο κατά την εκτέλεση του αλγορίθμου συμβαίνει ένας αριθμός οπισθοδρομήσεων. Πιο συγκεκριμένα, οπισθοδρομήσεις συμβαίνουν όταν φθάνουμε σε κορυφές από όπου δεν είναι δυνατόν να συνεχίσουμε (δηλαδή τις κορυφές 2, 4, 6, 12 και 13).

Ο αλγόριθμος Αναζήτησης κατά βάθος σε ένα γράφο είναι ένας γενικός αλγόριθμος και αποτελεί βασικό συστατικό πολλών συνθετότερων αλγορίθμων που σχετίζονται με γράφους. Έτσι, αυτός καθ’αυτός ο αλγόριθμος όπως διατυπώθηκε προηγουμένως δεν δίνει τη λύση στο πρόβλημα της διάσχισης του λαβύρινθου, αλλά αποτελεί τη βάση για τη διατύπωση ενός ειδικού αλγορίθμου. Πρακτικά, αν κάποιος πρέπει να διασχίσει ένα λαβύρινθο, πρέπει να τοποθετεί διάφορα σημάδια όταν φθάνει σε μία διασταύρωση, ώστε να γνωρίζει

- ⇒ αν έχει ξαναπεράσει τους διαδρόμους που ξεκινούν από τη διασταύρωση αυτή, και μάλιστα κατά την ίδια φορά που πιθανώς ακολουθήσει τώρα, και
- ⇒ ποιος είναι ο διάδρομος απ’ όπου για πρώτη φορά έφθασε στη συγκεκριμένη διασταύρωση, ώστε από εκεί να επιστρέψει προς την έξοδο.

Έτσι, ακολουθώντας αυτήν την τεχνική, κάποιος δεν θα χαθεί μέσα στο λαβύρινθο αλλά με βεβαιότητα θα διασχίσει κάθε διάδρομο του λαβυρίνθου δύο μόνο φορές, από μία κατά τις δύο κατευθύνσεις.