

ΒΑΣΙΚΕΣ ΕΠΕΞΕΡΓΑΣΙΕΣ ΜΟΝΟΔΙΑΣΤΑΤΩΝ ΚΑΙ ΔΙΣΔΙΑΣΤΑΤΩΝ
ΠΙΝΑΚΩΝ
ΟΙ ΠΙΟ ΣΗΜΑΝΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ

<u>ΕΥΡΕΣΗ ΜΕΓΑΛΥΤΕΡΟΥ/ΜΙΚΡΟΤΕΡΟΥ</u> <u>ΣΤΟΙΧΕΙΟΥ ΜΟΝΟΔΙΑΣΤΑΤΟΥ -1</u>	<u>ΕΥΡΕΣΗ ΜΕΓΑΛΥΤΕΡΟΥ/ΜΙΚΡΟΤΕΡΟΥ</u> <u>ΣΤΟΙΧΕΙΟΥ ΜΟΝΟΔΙΑΣΤΑΤΟΥ -2</u>
<p>Όλα τα στοιχεία του πίνακα είναι <u>διαφορετικά</u> μεταξύ τους. Βρίσκουμε το <u>μικρότερο/μεγαλύτερο</u> στοιχείο και την θέση στην οποία βρίσκεται.</p> <p>$MIN \leftarrow A[1]$ $MAX \leftarrow A[1]$ $P_{MAX} \leftarrow 1$ $P_{MIN} \leftarrow 1$</p> <p>Για I από 2 μέχρι N Αν $A[I] < MIN$ $MIN \leftarrow A[I]$ $P_{MIN} \leftarrow I$</p> <p>Τέλος_αν Αν $A[I] > MAX$ $MAX \leftarrow A[I]$ $P_{MAX} \leftarrow I$</p> <p>Τέλος_αν Τέλος_επανάληψης Γράψε 'Το μικρότερο στοιχείο είναι το:',MIN,'στη θέση',P_{MIN} Γράψε 'Το μεγαλύτερο στοιχείο είναι το :',MAX,'στη θέση',P_{MAX}</p>	<p>Δεν είναι όλα τα στοιχεία του πίνακα <u>διαφορετικά</u> μεταξύ τους.Βρίσκουμε το <u>μικρότερο/ μεγαλύτερο</u> στοιχείο, τις θέσεις που τα συναντάμε καθώς και το <u>πλήθος θέσεων που τα συναντάμε</u>.</p> <p>$MIN \leftarrow A[1]$ $MAX \leftarrow A[1]$ $pl_{min} \leftarrow 0$ $pl_{max} \leftarrow 0$</p> <p>Για I από 2 μέχρι N Αν $A[I] < MIN$ $MIN \leftarrow A[I]$ Τέλος_αν Αν $A[I] > MAX$ $MAX \leftarrow A[I]$ Τέλος_αν Τέλος_επανάληψης Για I από 1 μέχρι N Αν $A[I] = MIN$ Γράψε 'Θέση ελάχιστου στοιχείου :',I $pl_{min} \leftarrow pl_{min} + 1$ Τέλος_αν Αν $A[I] = MAX$ Γράψε 'Θέση μέγιστου στοιχείου :',I $pl_{max} \leftarrow pl_{max} + 1$ Τέλος_αν Τέλος_επανάληψης Γράψε 'Το μικρότερο στοιχείο', MIN,' βρέθηκε σε :',pl_{min}, 'θέσεις' Γράψε 'Το μεγαλύτερο στοιχείο',MAX,'βρέθηκε σε :',pl_{max}, 'θέσεις'</p>

<p><u>ΕΥΡΕΣΗ ΜΕΓ/ΜΙΚΡ ΣΤΟΙΧΕΙΟΥ ΔΙΣΔΙΑΣΤΑΤΟΥ ΠΙΝΑΚΑ ΚΑΙ ΘΕΣΕΩΝ</u></p> <p>$\max \leftarrow \text{ΠΙΝ}[1,1]$ $\min \leftarrow \text{ΠΙΝ}[1,1]$ $\Theta_imax \leftarrow 1$ $\Theta_jmax \leftarrow 1$ $\Theta_imin \leftarrow 1$ $\Theta_jmin \leftarrow 1$ Για i από 1 μέχρι N Για j από 1 μέχρι M Αν $\text{ΠΙΝ}[i,j] > \max$ τότε $\max \leftarrow \text{ΠΙΝ}[i,j]$ $\Theta_imax \leftarrow i$ $\Theta_jmax \leftarrow j$ Τέλος_αν Αν $\text{ΠΙΝ}[i,j] < \min$ τότε $\min \leftarrow \text{ΠΙΝ}[i,j]$ $\Theta_imin \leftarrow i$ $\Theta_jmin \leftarrow j$ Τέλος_αν Τέλος_επανάληψης Τέλος_επανάληψης</p>	<p><u>ΕΥΡΕΣΗ ΜΑΧ ΣΤΟΙΧΕΙΟΥ ΑΝΑ ΓΡΑΜΜΗ ΚΑΙ ΑΡΙΘΜΟΣ ΣΤΗΛΗΣ</u></p> <p>Για i από 1 μέχρι N $\max \leftarrow \text{ΠΙΝ}[i,1]$ $\Theta_jmax \leftarrow 1$ Για j από 2 μέχρι M Αν $\text{ΠΙΝ}[i,j] > \max$ τότε $\max \leftarrow \text{ΠΙΝ}[i,j]$ $\Theta_jmax \leftarrow j$ Τέλος_αν Τέλος_επανάληψης ! ΕΔΩ Η ΕΜΦΑΝΙΣΗ Τέλος_επανάληψης</p> <p><u>ΕΥΡΕΣΗ ΜΙΝ ΣΤΟΙΧΕΙΟΥ ΑΝΑ ΣΤΗΛΗ ΚΑΙ ΑΡΙΘΜΟΣ ΓΡΑΜΜΗΣ</u></p> <p>Για j από 1 μέχρι M $\min \leftarrow \text{ΠΙΝ}[1,j]$ $\Theta_imin \leftarrow 1$ Για i από 2 μέχρι N Αν $\text{ΠΙΝ}[i,j] < \min$ τότε $\Theta_imin \leftarrow i$ $\min \leftarrow \text{ΠΙΝ}[i,j]$ Τέλος_αν Τέλος_επανάληψης ! ΕΔΩ Η ΕΜΦΑΝΙΣΗ Τέλος_επανάληψης</p>
<p><u>ΣΕΙΡΙΑΚΗ ΑΝΑΖΗΤΗΣΗ ΣΤΟΙΧΕΙΟΥ ΣΕ ΜΟΝΟΔΙΑΣΤΑΤΟ ΠΙΝΑΚΑ</u></p> <p>a) Τα στοιχεία του πίνακα είναι διαφορετικά μεταξύ τους. <u>Βρίσκουμε το στοιχείο και την θέση στην οποία βρίσκεται (βιβλίο μαθητή)</u></p> <p>b) Τα στοιχεία του πίνακα δεν είναι διαφορετικά μεταξύ τους. Βρίσκουμε το στοιχείο και τις θέσεις στις οποίες βρίσκεται Για i από 1 μέχρι N Αν $\text{table}[i] = \text{key}$ Εμφάνισε "Θέση στοιχείου:", i Τέλος_αν Τέλος_επανάληψης</p> <p>c) Ο πίνακας είναι <u>ταξινομημένος</u> (και τα στοιχεία είναι διαφορετικά μεταξύ τους), άρα η αναζήτηση πρέπει να σταματήσει όταν βρεθεί</p>	

<p>στοιχείο μεγαλύτερο από αυτό που αναζητείται!</p> <p>..... Done ← Ψευδής Position ← 0 I ← 1 Όσο I ≤ N ΚΑΙ Done = Ψευδής επανάλαβε Αν table[I] = key τότε Done ← Αληθής Position ← I αλλιώς_αν table[I] > key τότε Done ← Αληθής αλλιώς I ← I + 1 Τέλος_αν Τέλος_επανάληψης Σε αυτή την λύση, η ένδειξη ότι το στοιχείο βρέθηκε είναι μόνο το Position <> 0.</p>	
<p><u>ΤΑΞΙΝΟΜΗΣΗ ΜΟΝΟΔΙΑΣΤΑΤΟΥ ΠΙΝΑΚΑ ΜΕ</u> <u>ΑΛΓΟΡΙΘΜΟ ΦΥΣΣΑΛΙΔΑΣ</u> <u>(ΠΛΗΘΟΣ ΣΤΟΙΧΕΙΩΝ = 10)</u></p> <p><u>1. ΑΥΞΟΥΣΑ ΣΕΙΡΑ ΤΑΞΙΝΟΜΗΣΗΣ</u></p> <p>ΓΙΑ I ΑΠΟ 2 ΜΕΧΡΙ 10 ΓΙΑ k ΑΠΟ 10 ΜΕΧΡΙ I ΜΕ_ΒΗΜΑ -1 Αν A[k-1] > A[k] τότε temp ← A[k] A[k] ← A[k-1] A[k-1] ← temp τέλος_αν ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ</p> <p>Ο δείκτης k διατρέχει το κομμάτι του πίνακα που δεν έχει ακόμη ταξινομηθεί. Δηλ. για κάθε τιμή του I (εξωτερικό ΓΙΑ), διατρέχεται ο πίνακας από το τέλος μέχρι I-1 (εσωτερικό ΓΙΑ). Σε κάθε τέτοιο πέρασμα του πίνακα, στην I-1 θέση μπαίνει το μικρότερο στοιχείο του περάσματος.</p> <p><u>2. ΦΘΙΝΟΥΣΑ ΣΕΙΡΑ ΤΑΞΙΝΟΜΗΣΗΣ</u></p> <p>ΓΙΑ I ΑΠΟ 2 ΜΕΧΡΙ 10 ΓΙΑ k ΑΠΟ 10 ΜΕΧΡΙ I ΜΕ_ΒΗΜΑ -1 Αν A[k-1] < A[k] τότε temp ← A[k] A[k] ← A[k-1] A[k-1] ← temp τέλος_αν ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ</p>	

<p>« ΕΞΥΠΝΟΣ» ΑΛΓΟΡΙΘΜΟΣ ΦΥΣΣΑΛΙΔΑΣ ΠΟΥ ΣΤΑΜΑΤΑ ΟΤΑΝ Ο ΠΙΝΑΚΑΣ ΤΑΞΙΝΟΜΗΘΕΙ, ΩΣΤΕ ΝΑ ΜΗΝ ΓΙΝΟΝΤΑΙ ΠΕΡΙΤΤΑ ΠΕΡΑΣΜΑΤΑ.</p> <p>Done ← Αληθής I ← 2 Όσο (I ≤ 10) ΚΑΙ (Done = Αληθής) επανάλαβε Done ← Ψευδής !Αν στο πέρασμα που ακολουθεί γίνουν αντιμεταθέσεις, η done γίνεται αληθής. Αν δεν γίνει αληθής, σημαίνει ότι στο πέρασμα δεν έγινε καμμία αντιμετάθεση, άρα ο πίνακας έχει ταξινομηθεί και δεν υπάρχει λόγος να συνεχιστούν τα επόμενα περάσματα. Για J από 10 μέχρι I με_βήμα -1 Αν A[J-1] > A[J] τότε TEMP ← A[J-1] A[J-1] ← A[J] A[J] ← TEMP Done ← Αληθής Τελος_αν Τέλος_επανάληψης I ← I+1 Τέλος_επανάληψης</p> <p>Ο αλγόριθμος αυτός (μια παραλλαγή του) δίνεται έτοιμος ως λυμένο παράδειγμα στο κεφάλαιο 10, όπου έχει υλοποιηθεί με υποπρόγραμμα. Να το διαβάσετε.</p>	
<p>ΣΕΙΡΙΑΚΗ ΑΝΑΖΗΤΗΣΗ ΣΤΟΙΧΕΙΟΥ ΣΕ ΔΙΣΔΙΑΣΤΑΤΟ ΠΙΝΑΚΑ (<u>Όλα τα στοιχεία του πίνακα διαφορετικά μεταξύ τους</u>)</p> <p>Done ← Ψευδής POSI ← 0 POSJ ← 0 I ← 1 Διάβασε key Όσο (I ≤ M) ΚΑΙ (Done = Ψευδής) επανάλαβε J ← 1 Όσο (J ≤ N) ΚΑΙ (Done = Ψευδής) επανάλαβε Αν A[I,J]=key τότε Done ← Αληθής POSI ← I POSJ ← J Αλλιώς J ← J+1 Τελος_αν Τέλος_επανάληψης I ← I+1 Τέλος_επανάληψης</p>	<p>ΤΑΞΙΝΟΜΗΣΗ ΣΕ ΔΙΣΔΙΑΣΤΑΤΟΥ ΜΕ ΤΟΝ ΑΛΓΟΡΙΘΜΟ ΤΗΣ ΦΥΣΣΑΛΙΔΑΣ</p> <p>ΣΤΗΛΗ ΔΙΣΔΙΑΣΤΑΤΟΥ ΠΙΝΑΚΑ (ΚΑΙ ΠΑΡΑΛΛΗΛΗ ΤΑΞΙΝΟΜΗΣΗ ΜΟΝΟΔΙΑΣΤΑΤΟΥ)</p> <p>ΕΣΤΩ ΕΝΑΣ ΜΟΝΟΔΙΑΣΤΑΤΟΣ ΠΙΝΑΚΑΣ ΜΕ 10 ΠΟΛΕΙΣ ΚΑΙ ΕΝΑΣ ΔΙΣΔΙΑΣΤΑΤΟΣ ΟΠΟΥ ΓΙΑ ΚΑΘΕ ΠΟΛΗ ΑΠΟΘΗΚΕΥΟΥΜΕ ΜΙΑ ΘΕΡΜΟΚΡΑΣΙΑ ΑΝΑ ΗΜΕΡΑ ΜΙΑΣ ΕΒΔΟΜΑΔΑΣ). ΘΕΛΟΥΜΕ ΝΑ ΕΜΦΑΝΙΣΟΥΜΕ ΤΙΣ ΠΟΛΕΙΣ ΤΑΞΙΝΟΜΗΜΕΝΕΣ ΚΑΤΑ ΑΥΞΟΥΣΑ ΘΕΡΜΟΚΡΑΣΙΑ 2^{ΗΣ} ΗΜΕΡΑΣ ΤΗΣ ΕΒΔΟΜΑΔΑΣ. ΘΑ ΤΑΞΙΝΟΜΗΣΟΥΜΕ ΤΗ 2^Η ΣΤΗΛΗ ΤΟΥ ΠΙΝΑΚΑ, ΚΑΙ ΤΑΥΤΟΧΡΟΝΑ ΤΟΝ ΠΙΝΑΚΑ ΟΝΟΜΑΤΩΝ. ΑΝΤΙΜΕΤΩΠΙΖΟΥΜΕ ΤΗ 2^Η ΣΤΗΛΗ ΤΟΥ ΘΕΡ ΩΣ ΜΟΝΟΔΙΑΣΤΑΤΟ ΠΙΝΑΚΑ 10 ΣΤΟΙΧΕΙΩΝ.</p>

	<p>ΓΙΑ Ι ΑΠΟ 2 ΜΕΧΡΙ 10 ΓΙΑ Κ ΑΠΟ 10 ΜΕΧΡΙ Ι ΜΕ_ΒΗΜΑ -1 ΑΝ ΘΕΡ[Κ,2]<ΘΕΡ[Κ-1,2] ΤΟΤΕ TEMP ← ΘΕΡ[Κ-1,2] ΘΕΡ[Κ-1,2] ← ΘΕΡ[Κ,2] ΘΕΡ[Κ,2] ← TEMP TEMP1 ← ΟΝ[Κ-1] ΟΝ[Κ-1] ← ΟΝ[Κ] ΟΝ[Κ] ← TEMP1 ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ! ΤΑΞΙΝΟΜΗΣΗ ΓΡΑΜΜΗΣ , ΠΧ. ΤΗΣ 2^{ΗΣ} . ΔΗΛΑΔΗ, ΘΕΛΟΥΜΕ ΝΑ ΤΑΞΙΝΟΜΗΣΟΥΜΕ ΤΙΣ 7 ΘΕΡΜΟΚΡΑΣΙΕΣ ΤΗΣ 2^{ΗΣ} ΠΟΛΗΣ (ΕΔΩ ΔΕΝ ΕΧΕΙ ΕΝΝΟΙΑ ΠΑΡΑΛΛΗΛΗ ΤΑΞΙΝΟΜΗΣΗ ΜΕ ΤΟΝ ΜΟΝΟΔΙΑΣΤΑΤΟ) ΓΙΑ Ι ΑΠΟ 2 ΜΕΧΡΙ 7 ΓΙΑ Κ ΑΠΟ 7 ΜΕΧΡΙ Ι ΜΕ_ΒΗΜΑ -1 ΑΝ ΘΕΡ[2,Κ]<ΘΕΡ[2,Κ-1] ΤΟΤΕ TEMP ← ΘΕΡ[2,Κ-1] ΘΕΡ[2,Κ-1] ← ΘΕΡ[2,Κ] ΘΕΡ[2,Κ] ← TEMP ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ! ΤΑΞΙΝΟΜΗΣΗ ΚΑΘΕ ΓΡΑΜΜΗΣ ΑΝΑ ΘΕΡΜΟΚΡΑΣΙΑ ΚΑΙ ΕΜΦΑΝΙΣΗ ΤΗΣ ΓΡΑΜΜΗΣ ΜΕΤΑ ΤΗΝ ΤΑΞΙΝΟΜΗΣΗ!!! ! ΤΟ Μ ΕΙΝΑΙ Ο ΔΕΙΚΤΗΣ ΤΗΣ ΓΡΑΜΜΗΣ ΓΙΑ Μ ΑΠΟ 1 ΜΕΧΡΙ 10 ! ΓΙΑ ΚΑΘΕ ΓΡΑΜΜΗ ΚΑΝΩ ΦΥΣΣΑΛΙΔΑ ΓΙΑ Ι ΑΠΟ 2 ΜΕΧΡΙ 7 ΓΙΑ Κ ΑΠΟ 7 ΜΕΧΡΙ Ι ΜΕ_ΒΗΜΑ -1 ΑΝ ΘΕΡ[Μ,Κ]<ΘΕΡ[Μ,Κ-1] ΤΟΤΕ TEMP ← ΘΕΡ[Μ,Κ-1] ΘΕΡ[Μ,Κ-1] ← ΘΕΡ[Μ,Κ] ΘΕΡ[Μ,Κ] ← TEMP ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ! ΕΜΦΑΝΙΣΗ ΤΗΣ ΓΡΑΜΜΗΣ ΜΕΤΑ ΤΗΝ ΤΑΞΙΝΟΜΗΣΗ ΓΙΑ Λ ΑΠΟ 1 ΜΕΧΡΙ 7 ΓΡΑΨΕ ΘΕΡ[Μ,Λ] ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ! ΠΑΜΕ ΓΙΑ ΤΗΝ ΕΠΟΜΕΝΗ ΓΡΑΜΜΗ.... ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ</p>

ΣΥΓΧΩΝΕΥΣΗ ΔΥΟ ΤΑΞΙΝΟΜΗΜΕΝΩΝ ΠΙΝΑΚΩΝ

A, B ΣΕ ΤΡΙΤΟ C.

(πλήθος στοιχείων A,B = 100)

...

! i ο δείκτης του A

! j ο δείκτης του B

! k ο δείκτης του C

i ← 1

j ← 1

k ← 1

Όσο i ≤ 100 και j ≤ 100 επανάλαβε

 Αν A[i] ≤ B[j] τότε

 C[k] ← A[i]

 i ← i + 1

 αλλιώς

 C[k] ← B[j]

 j ← j + 1

 τέλος_αν

 k ← k + 1

τέλος_επανάληψης

! Αν τελείωσαν τα στοιχεία του A, βάλτε όσα
απέμειναν από τον B

Αν i > 100 τότε

 Όσο (j ≤ 100) επανάλαβε

 C[k] ← B[j]

 j ← j + 1

 k ← k + 1

 τέλος_επανάληψης

αλλιώς

! Αν τελείωσαν τα στοιχεία του B, βάλτε όσα
απέμειναν από τον A

 Όσο (i ≤ 100) επανάλαβε

 C[k] ← A[i]

 i ← i + 1

 k ← k + 1

 τέλος_επανάληψης

τέλος_αν

...

<p>ΥΠΟΛΟΓΙΣΜΟΣ ΑΘΡΟΙΣΜΑΤΟΣ / ΜΟ ΑΝΑ ΓΡΑΜΜΗ ΚΑΙ ΣΤΗΛΗ ΔΙΣΔΙΑΣΤΑΤΟΥ ΤΑΥΤΟΧΡΟΝΑ</p> <p><i>(ΔΗΜΙΟΥΡΓΙΑ πινάκων row[N] και col[M]. Στο row[i] αποθηκεύεται ο ΜΟ της γραμμής i. Στο col[j] αποθηκεύεται ο ΜΟ της στήλης j)</i></p> <p>! Αρχικοποίηση των πινάκων Για I από 1 μέχρι N row[I] ← 0 Τέλος_επανάληψης Για I από 1 μέχρι M col[I] ← 0 Τέλος_επανάληψης</p> <p>!Υπολογισμός και αποθήκευση αθροίσματος ανά γραμμή και στήλη ταυτόχρονα Για I από 1 μέχρι N Για j από 1 μέχρι M row[I] ← row[I] + ΠΙΝ[I,j] col [j] ← col[j] + ΠΙΝ [I,j] Τέλος_επανάληψης Τέλος_επανάληψης</p> <p>!Υπολογισμός και αποθήκευση ΜΟ ανά γραμμή και στήλη Για I από 1 μέχρι N row[I] ← row[I] / M Τέλος_επανάληψης Για I από 1 μέχρι M col[I] ← col[I] / N Τέλος_επανάληψης</p>	

Επισημαίνεται ότι οι πίνακες στο βιβλίο της Β' τάξης αντιμετωπίζονται ως δυναμικές δομές, ενώ στο βιβλίο της Γ' τάξης ορίζονται ως στατικές δομές. Συνεπώς για τη Γ' τάξη και τη ΓΛΩΣΣΑ, η δομή του πίνακα είναι στατική και για να χρησιμοποιηθεί ένας πίνακας θα πρέπει να έχει πρώτα δηλωθεί, τόσο ο πίνακας, όσο και το μέγεθός του. Επίσης και οι δομές ουρά και στοίβα θεωρούνται στατικές δομές για τη ΓΛΩΣΣΑ, επειδή υλοποιούνται με πίνακες.

Γ) ΔΥΑΔΙΚΗ ΑΝΑΖΗΤΗΣΗ

Ο αλγόριθμος της δυαδικής αναζήτησης (binary search) εφαρμόζεται μόνο σε πίνακες που έχουν ταξινομημένα στοιχεία. Αν τα στοιχεία δεν είναι ταξινομημένα τότε δεν μπορεί να εφαρμοστεί.

Ο αλγόριθμος λειτουργεί ως εξής:

Βρίσκουμε το μεσαίο στοιχείο του ταξινομημένου πίνακα.

Εάν το προς αναζήτηση στοιχείο είναι ίσο με το μεσαίο στοιχείο τότε σταματάμε την αναζήτηση αφού το στοιχείο βρέθηκε

Εάν δεν βρέθηκε, τότε ελέγχουμε αν το στοιχείο που αναζητούμε είναι μικρότερο ή μεγαλύτερο από το μεσαίο στοιχείο του πίνακα. Αν είναι μικρότερο, περιορίζουμε την αναζήτηση στο πρώτο μισό του πίνακα (με την προϋπόθεση ότι τα στοιχεία είναι διατεταγμένα κατά αύξουσα σειρά), ενώ αν είναι μεγαλύτερο περιορίζουμε την αναζήτηση στο δεύτερο μισό του πίνακα.

Η διαδικασία αυτή λοιπόν επαναλαμβάνεται για το κατάλληλο πρώτο ή δεύτερο μισό πίνακα, μετά για το 1/4 του πίνακα κ.ο.κ. μέχρι, είτε να βρεθεί το στοιχείο, είτε να μην είναι δυνατό να χωρισθεί ο πίνακας περαιτέρω σε δύο νέα μέρη.

Αλγόριθμος δυαδικής αναζήτησης

αλγόριθμος Δυαδική_αναζήτηση !Α μονοδιάστατος πίνακας N θέσεων, S το στοιχείο που αναζητούμε

δεδομένα // N, A, S //

Left ← 1 ! αριστερό όριο

Right ← N ! δεξιό όριο

K ← 0 ! θέση του στοιχείου

F ← FALSE

όσο (Left<=Right) και (f=FALSE) **επανάλαβε**

M ← (Left+Right) div 2

αν A[M]=S **τότε**

K ← M;

F ← TRUE;

αλλιώς

αν $A[M] < S$ τότε

Left $\leftarrow M+1$;

αλλιώς

Right $\leftarrow M-1$;

Τέλος_αν

Τέλος_αν

Τέλος_επανάληψης

Αν $F = \text{TRUE}$ τότε

Εμφάνισε "Το στοιχείο", S, "υπάρχει στη θέση:", M

Αλλιώς

Εμφάνισε "Το στοιχείο", S, " δεν υπάρχει στον πίνακα"

Τέλος_αν

Η δυαδική αναζήτηση να διδαχθεί ως άσκηση και να υλοποιηθεί με πρόγραμμα, όπως παρακάτω σε ταξινομημένο πίνακα 20 θέσεων. Πέρα από το τμήμα δηλώσεων, το πρόγραμμα έχει ένα επιπλέον τμήμα για το "γέμισμα" του πίνακα με στοιχεία (υποθέτουμε ότι ο πίνακας γεμίζει με σωστά ταξινομημένα στοιχεία σε αύξουσα σειρά).

ΠΡΟΓΡΑΜΜΑ δυαδική_αναζήτηση

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[20], Left, Right, M, k, S, i

ΛΟΓΙΚΕΣ: f

ΑΡΧΗ

ΓΡΑΨΕ 'Οι αριθμοί που θα δοθούν πρέπει να είναι ταξινομημένοι κατά αύξουσα τάξη'

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 20

ΓΡΑΨΕ 'Δώσε το', i, ' στοιχείο του πίνακα'

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Δώσε τιμή για αναζήτηση: '

ΔΙΑΒΑΣΕ S

Left \leftarrow 1

Right \leftarrow 20

k \leftarrow 0

f \leftarrow ΨΕΥΔΗΣ

ΟΣΟ (Left \leq Right) **ΚΑΙ** (f = ΨΕΥΔΗΣ) **ΕΠΑΝΑΛΑΒΕ**

M \leftarrow (Left + Right) DIV 2

ΑΝ A[M] = S **ΤΟΤΕ**

k \leftarrow M

f \leftarrow ΑΛΗΘΗΣ

ΑΛΛΙΩΣ

ΑΝ $A[M] < S$ **ΤΟΤΕ**

Left $\leftarrow M + 1$

ΑΛΛΙΩΣ

Right $\leftarrow M - 1$

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ $f = \text{ΑΛΗΘΗΣ}$ **ΤΟΤΕ**

ΓΡΑΨΕ "Το στοιχείο, " S , "υπάρχει στη θέση:", M

ΑΛΛΙΩΣ

ΓΡΑΨΕ "Το στοιχείο, " S , " δεν υπάρχει στον πίνακα"

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ δυαδική_αναζήτηση

Παράδειγμα

Δίνεται ο πίνακας

1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
---	---	---	---	---	----	----	----	----	----	----	----	----	----	----

Αναζήτηση του στοιχείου 38 (υπάρχει στον πίνακα)

Βήμα 1	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 2	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 3	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 4	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47

Με κίτρινο σημειώνεται το στοιχείο του πίνακα που εξετάζεται (στο μέσον)

Με πράσινο σημειώνεται το τμήμα του πίνακα που απομένει για αναζήτηση

Με κόκκινο σημειώνεται το τμήμα του πίνακα που έχει αποκλειστεί

(Τα ίδια χρώματα χρησιμοποιούνται και στο επόμενο παράδειγμα)

Αναζήτηση του στοιχείου 39 (δεν υπάρχει στον πίνακα)

Βήμα 1	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 2	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 3	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 4	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 5	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47

Αριθμός συγκρίσεων στη δυαδική αναζήτηση

Στοιχεία N	Συγκρίσεις
10	4
100	7
1.000	10
10.000	14
100.000	17
1.000.000	20
10.000.000	24
100.000.000	27
1.000.000.000	30

*Ως άσκηση μπορεί να δοθεί η βελτιστοποίηση του αλγορίθμου δυαδικής αναζήτησης έτσι ώστε να επιτρέπει διαδοχικές αναζητήσεις πολλών στοιχείων. Η αναζήτηση να τερματίζεται όταν δοθεί κάποιος συγκεκριμένος αριθμός ή με ερώτηση "Θέλετε άλλη αναζήτηση (N/O)"

Δ) ΤΑΞΙΝΟΜΗΣΗ ΜΕ ΕΠΙΛΟΓΗ (SELECTION SORT)

Η ταξινόμηση με επιλογή (selection sort), αποτελεί βασικό τρόπο ταξινόμησης, που υλοποιείται σε ένα μονοδιάστατο πίνακα σε τρία βήματα:

1. Επιλογή του ελάχιστου στοιχείου
2. Ανταλλαγή του ελάχιστου με το πρώτο στοιχείο
3. Επανάληψη των βημάτων 1 και 2 για τα υπόλοιπα στοιχεία του πίνακα

Ο Αλγόριθμος ταξινόμησης με επιλογή είναι ο παρακάτω.

Αλγόριθμος Selection_Sort

Δεδομένα // table, n //

Για i από 1 μέχρι n-1

$k \leftarrow i$

$x \leftarrow \text{table}[i]$

Για j από i+1 μέχρι n

Αν $x > \text{table}[j]$ Τότε

$k \leftarrow j$

$x \leftarrow \text{table}[j]$

Τέλος_Επανάληψης

$\text{table}[k] \leftarrow \text{table}[i]$

$\text{table}[i] \leftarrow x$

Τέλος_επανάληψης

Η υλοποίηση του αλγορίθμου ταξινόμησης με επιλογή, να διδαχθεί ως άσκηση και να υλοποιηθεί με πρόγραμμα όπως παρακάτω. Πέρα από το τμήμα δηλώσεων, το πρόγραμμα έχει δύο επιπλέον τμήματα, ένα τμήμα για το "γέμισμα" του πίνακα με στοιχεία και ένα τμήμα για την εκτύπωση του ταξινομημένου πίνακα.

ΠΡΟΓΡΑΜΜΑ Selection_Sort

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[20], K1, x, i, j

ΑΡΧΗ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 20

ΓΡΑΨΕ 'Δώσε το', i, ' στοιχείο του πίνακα'

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 19

K1 <- i

x <- A[i]

ΓΙΑ j **ΑΠΟ** i + 1 **ΜΕΧΡΙ** 20

ΑΝ x > A[j] **ΤΟΤΕ**

K1 <- j

x <- A[j]

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

A[K1] <- A[i]

A[i] <- x

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Εκτύπωση με ταξινομημένα τα στοιχεία'

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 20

ΓΡΑΨΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Selection_Sort

Παράδειγμα

Αν υποθέσουμε ότι έχουμε το πίνακα A[8] με στοιχεία τους αριθμούς 46, 55, 12, 42, 94, 18, 06, 67. Δηλαδή σε μορφή μονοδιάστατου πίνακα:

46	55	12	42	94	18	06	67
----	----	----	----	----	----	----	----

τότε παρακάτω φαίνεται πως μετακινούνται τα στοιχεία με τον αλγόριθμο SelectionSort

Βήμα 1 (εύρεση του ελάχιστου των στοιχείων και ανταλλαγή με το πρώτο)

46	55	12	42	94	18	06	67
----	----	----	----	----	----	----	----

Βήμα 2 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το δεύτερο στοιχείο και κάτω)

06	55	12	42	94	18	46	67
----	----	----	----	----	----	----	----

Βήμα 3 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το τρίτο στοιχείο και κάτω)

06	12	55	42	94	18	46	67
----	----	----	----	----	----	----	----

Βήμα 4 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το τέταρτο στοιχείο και κάτω)

06	12	18	42	94	55	46	67
----	----	----	----	----	----	----	----

Βήμα 5 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το πέμπτο στοιχείο και κάτω)

06	12	18	42	94	55	46	67
----	----	----	----	----	----	----	----

Βήμα 6 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το έκτο στοιχείο και κάτω)

06	12	18	42	46	55	94	67
----	----	----	----	----	----	----	----

Βήμα 7 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το έβδομο στοιχείο και κάτω)

06	12	18	42	46	55	94	67
----	----	----	----	----	----	----	----

Τελική μορφή ταξινομημένου πίνακα (δεν χρειάζεται 8^η επανάληψη σύγκρισης, αφού όταν απομένουν δύο μόνο κελιά και στο πρώτο θέσεις τον μικρότερο αριθμό, τότε στο δεύτερο αναγκαστικά τίθεται ο μεγαλύτερος)

06	12	18	42	46	55	67	94
----	----	----	----	----	----	----	----

Ε) ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΘΕΩΡΗΤΙΚΕΣ ΑΣΚΗΣΕΙΣ ΣΕ ΣΤΟΙΒΑ ΚΑΙ ΟΥΡΑ

- 1) Σε μια στοίβα 10 θέσεων έχουν τοποθετηθεί διαδοχικά τα στοιχεία: Σ, Γ, Μ, Α, Δ στην 1η, 2η, 3η, 4η και 5η θέση αντίστοιχα.
 - i) Να προσδιορίσετε την τιμή του δείκτη **top** της παραπάνω στοίβας και να την σχεδιάσετε.
 - ii) Αν εφαρμόσουμε τις παρακάτω λειτουργίες: **Απόθεση, Απόθεση, Απόθεση, Ώθηση Χ, Ώθηση Δ** και **Απόθεση** ποιά είναι η νέα τιμή της **top** και ποιά η τελική μορφή της στοίβας;
- 2) Η παραπάνω άσκηση να υλοποιηθεί με ουρά χρησιμοποιώντας, όπου Απόθεση **Εξαγωγή** και όπου Ώθηση **Εισαγωγή**. Επίσης αντί της top να δοθούν οι τιμές των δεικτών rear και front.
- 3) Σε μια άδεια στοίβα 10 θέσεων ωθούμε τα στοιχεία Ο, Σ, Ε, Τ, Λ. Με ποίο τρόπο πρέπει να ωθηθούν και να απωθηθούν τα δεδομένα ώστε η στοίβα να περιέχει τα δεδομένα Τ, Ε, Λ, Ο, Σ (σε αύξουσες θέσεις του πίνακα).
- 4) Η παραπάνω άσκηση να υλοποιηθεί με ουρά, χρησιμοποιώντας όπου Απόθεση **Εξαγωγή** και όπου Ώθηση **Εισαγωγή**. Επίσης αντί της top να δοθούν οι τιμές των δεικτών rear και front.