

Οδηγός μελέτης – Τυπικές επεξεργασίες πινάκων

(Θ Κεφ. 3.3, 9.4) (Ο Κεφ. 4.3, 4.4) (Π Κεφ. 2)

1. Τυπικές επεξεργασίες πινάκων

Έχοντας ξεκινήσει, από το προηγούμενο φυλλάδιο, να δουλεύουμε με τους πίνακες θα συνεχίσουμε να αναλύουμε σταδιακά τις πέντε πιο συχνές επεξεργασίες :

- ✓ Υπολογισμός του αθροίσματος των στοιχείων του πίνακα
- ✓ Εύρεση του μέγιστου ή ελάχιστου στοιχείου
- ✓ Ταξινόμηση των στοιχείων του πίνακα
- ✓ Αναζήτηση ενός στοιχείου του πίνακα
- ✓ Συγχώνευση δύο πινάκων.

2. 1^η Λειτουργία - Υπολογισμός του αθροίσματος των στοιχείων ενός πίνακα

Την επεξεργασία των στοιχείων ενός πίνακα με την προσπέλαση του την έχουμε παρουσιάσει και αναλύσει στην εισαγωγική ενότητα 13!

3. 2^η Λειτουργία - Εύρεση του μέγιστου ή του ελάχιστου στοιχείου

Εύρεση του μέγιστου και της θέσης του αν είναι μοναδικό	Εύρεση του μέγιστου και της θέσης του αν ΔΕΝ είναι μοναδικό
<pre> max <-- Π[1] θ <-- 1 Για I από 2 μέχρι 50 Αν Π[I] > max τότε max <-- I θ <-- I Τέλος_αν Τέλος_επανάληψης Γράψε max, θ </pre>	<pre> max <-- Π[1] Για I από 2 μέχρι 50 Αν Π[I] > max τότε max <-- I Τέλος_αν Τέλος_επανάληψης Γράψε max Για I από 1 μέχρι 50 Αν Π[I] = max τότε Γράψε I Τέλος_αν Τέλος_επανάληψης </pre>

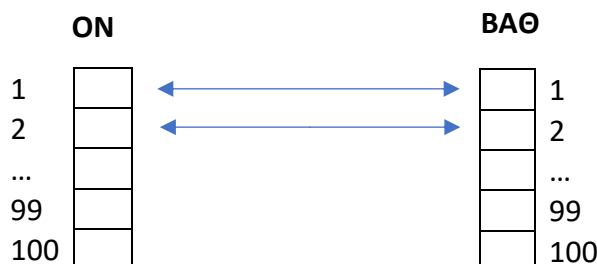
4. Παράλληλοι Πίνακες

Όταν έχουμε εγγραφές με πεδία που έχουν διαφορετικού τύπου δεδομένα και τις οποίες θέλουμε να καταχωρήσουμε σε πίνακες, δημιουργούμε για κάθε πεδίο ένα διαφορετικό πίνακα. Οι πίνακες αυτοί πρέπει να συνδέονται μεταξύ τους με τέτοιο τρόπο, ώστε η κάθε εγγραφή να έχει όλα τα στοιχεία της με την ίδια τιμή δείκτη, δηλ. να βρίσκονται στην ίδια θέση στους πίνακες.

Παράδειγμα 1

«Εστω ότι θέλουμε να αποθηκεύσουμε τα ονόματα και τον τελικό βαθμό 100 μαθητών του σχολείου μας!»

Για κάθε εγγραφή (μαθητή) χρειαζόμαστε να αποθηκεύσουμε ξεχωριστά το όνομα (τύπος χαρακτήρα) και τον τελικό βαθμό (τύπος πραγματικός αριθμός) οπότε χρησιμοποιούμε δύο πίνακες 100 θέσεων.



Αυτό που πρέπει να προσέξω κατά την διαδικασία της εισαγωγής των στοιχείων στους πίνακες είναι να αποθηκεύω τα δεδομένα ενός μαθητή στην ίδια θέση και στους δύο πίνακες. Οπότε αν αποθηκεύσω το όνομα του μαθητή Γεωργουλάκης Γιώργος στην πρώτη θέση του πίνακα **ON** τότε πρέπει στην αντίστοιχη πρώτη θέση του πίνακα **BAΘ** να αποθηκεύσω το τελικό βαθμό του!

Με αυτό τον τρόπο γνωρίζω πάντα ότι ο μαθητής που το όνομά του είναι αποθηκευμένο στην 4^η θέση του πίνακα **ON[4]** έχει τον βαθμό του αποθηκευμένο στην 4^η θέση του δεύτερου πίνακα **BAΘ[4]**.

Εισαγωγή στοιχείων	
Χωρίς έλεγχο δεδομένων	Με έλεγχο δεδομένων
<pre> ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 100 ΓΡΑΨΕ 'ΔΩΣΕ ΟΝΟΜΑ ΚΑΙ ΒΑΘΜΟ' ΔΙΑΒΑΣΕ ON[I],BAΘ[I] ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ </pre>	<pre> ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 100 ΓΡΑΨΕ 'ΔΩΣΕ ΟΝΟΜΑ ΚΑΙ ΒΑΘΜΟ' ΔΙΑΒΑΣΕ ON[I] ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ ΔΙΑΒΑΣΕ BAΘ[I] ΜΕΧΡΙΣ_ΟΤΟΥ BAΘ[I]>=1 ΚΑΙ BAΘ[I]<=20 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ </pre>

Ας υποθέσουμε ότι στην συνέχεια το ερώτημα της άσκησης είναι να βρούμε τον μαθητή που είχε τον μεγαλύτερο βαθμό. **Προσοχή!** αφού δεν μας λέει ότι οι τιμές στον πίνακα BAΘ είναι μοναδικές πρέπει να υποθέσουμε ότι μπορεί να υπάρχουν περισσότεροι μαθητές που έχουν τον ίδιο βαθμό.

```

MAX <-- BAΘ[1]
ΓΙΑ I ΑΠΟ 2 ΜΕΧΡΙ 100
    ΑΝ BAΘ[I] > MAX ΤΟΤΕ
        MAX <-- BAΘ[I]
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 100
    ΑΝ BAΘ[I] = MAX ΤΟΤΕ
        ΓΡΑΨΕ ON[I]
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    
```

Αν εντοπίσουμε την μέγιστη τιμή στην θέση I τότε στην αντίστοιχη θέση στον παράλληλο πίνακα **ON[I]** έχει αποθηκευτεί το όνομα του μαθητή με αυτό τον βαθμό!

5. 3^η Λειτουργία - Αναζήτηση ενός στοιχείου του πίνακα.

Η Σειριακή αναζήτηση

Με τη μέθοδο αυτή ξεκινάμε από την αρχή του πίνακα και συγκρίνουμε κάθε τιμή του με το στοιχείο που αναζητάμε. Η σειριακή αναζήτηση είναι η πιο απλή, αλλά και η λιγότερο αποτελεσματική μέθοδος.

Η χρήση της δικαιολογείται μόνο στις παρακάτω περιπτώσεις:

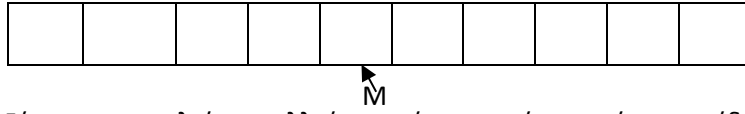
- ✓ Ο πίνακας δεν είναι ταξινομημένος.
- ✓ Ο πίνακας έχει μικρό μέγεθος.
- ✓ Η αναζήτηση στον πίνακα γίνεται σπάνια.

```

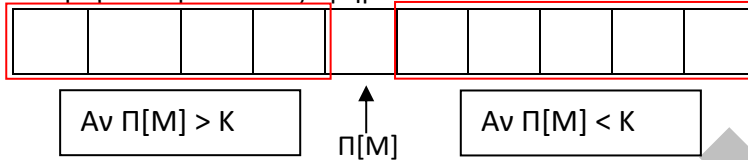
Pos <-- 0
I <-- 1
ΟΣΟ I<=50 ΚΑΙ Pos=0 ΕΠΑΝΑΛΑΒΕ
    ΑΝ Π[I] = K ΤΟΤΕ
        Pos <-- I
    ΤΕΛΟΣ_ΑΝ
    I <-- I + 1
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΝ Pos=0 ΤΟΤΕ
    ΓΡΑΨΕ 'Δεν υπάρχει'
ΑΛΛΙΩΣ
    ΓΡΑΨΕ 'Στην θέση',Pos
ΤΕΛΟΣ_ΑΝ
    
```

Η δυαδική αναζήτηση

Η δυαδική αναζήτηση είναι αποτελεσματικότερη από τη σειριακή, διότι χρειάζεται πολύ λιγότερο χρόνο εκτέλεσης όμως εφαρμόζεται μόνο σε πίνακες που έχουν ταξινομημένα στοιχεία.
Έστω ένας πίνακας που είναι ταξινομημένος κατά αύξουσα σειρά. Η βασική ιδέα είναι ότι βρίσκω το μεσαίο στοιχείο του πίνακα και το συγκρίνω με το στοιχείο που αναζητώ.



Αν είναι το ίδιο τότε η αναζήτηση μου τελείωσε αλλιώς αν είναι μικρότερο κάνω την ίδια διαδικασία στο αριστερό τμήμα του πίνακα ενώ αν είναι μεγαλύτερο στο δεξί τμήμα.



Αυτό συνεχίζετε μέχρι να βρούμε το στοιχείο ή το Αριστερό άκρο γίνει μεγαλύτερο από το Δεξί άκρο του πίνακα οπότε σημαίνει ότι το στοιχείο δεν υπάρχει στο πίνακα.

```

ΑΛΟΓΟΡΙΘΜΟΣ Δυαδική_αναζήτηση
Δεδομένα //Π,Ν,Κ//
Δ ← Ν           ! Δεξιό άκρο
Α ← 1           ! Αριστερό άκρο
Φ ← 0           ! η θέση του στοιχείου αν βρεθεί
ΟΣΟ Α<=Δ και Φ=0 ΕΠΑΝΑΛΑΒΕ
  Μ ← (Δ+Α) div 2
  ΑΝ Π[ Μ ] = Κ ΤΟΤΕ
    Φ ← Μ
  ΑΛΛΙΩΣ ΑΝ Π[ Μ ] > Κ ΤΟΤΕ
    Δ ← Μ - 1
  ΑΛΛΙΩΣ
    Α ← Μ + 1
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
Αποτελέσματα //Φ// ! Αν είναι 0 δεν έχει βρεθεί
ΤΕΛΟΣ Δυαδική_αναζήτηση
    
```

Παράδειγμα λειτουργίας: Αναζήτηση της τιμής του 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

Αναζήτηση της τιμής 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

Ο μέγιστος αριθμός των συγκρίσεων (επαναλήψεων) που απαιτούνται για την εύρεση ενός στοιχείου με την μέθοδο της δυαδικής αναζήτησης σε ένα σύνολο «n» ταξινομημένων στοιχείων, συμπεριλαμβανομένης και της περίπτωσης μη ύπαρξης του στοιχείου, δίνεται από το ακέραιο μέρος του $[\log_2(n)+1]$.

6. Μέθοδος Διαίρει και Βασίλευε. (Βιβλίο Πληροφορικής Κεφ. 2)

Επανάληψη από το φυλλάδιο 2 (Βιβλίο ΑΕΠΠ - Κεφάλαιο 4.1)

Υπάρχουν "συγγενή" προβλήματα, δηλαδή προβλήματα που μπορούν να αναλυθούν με παρόμοιο τρόπο και να αντιμετωπισθούν με αντίστοιχες μεθόδους και τεχνικές. Οι μέθοδοι ανάλυσης και επίλυσης των προβλημάτων παρουσιάζουν ιδιαίτερο ενδιαφέρον για τους εξής λόγους:

- παρέχουν ένα γενικό πρότυπο κατάλληλο για την επίλυση προβλημάτων ευρείας κλίμακας,
- μπορούν να αναπαρασταθούν με κοινές δομές δεδομένων και ελέγχου (που υποστηρίζονται από τις περισσότερες σύγχρονες γλώσσες προγραμματισμού),
- παρέχουν τη δυνατότητα καταγραφής των χρονικών και «χωρικών» απαιτήσεων της μεθόδου επίλυσης, έτσι ώστε να μπορεί να γίνει επακριβής εκτίμηση των αποτελεσμάτων.

Μία από αυτές τις μεθόδους που χρησιμοποιήσαμε για τον αλγόριθμο της δυαδικής αναζήτησης είναι η Μέθοδος «Διαίρει και Βασίλευε»



Η «**Διαίρει και Βασίλευε**» (divide and conquer) αποτελεί μια μέθοδο σχεδίασης αλγορίθμων στην οποία εντάσσονται οι τεχνικές που υποδιαιρούν ένα πρόβλημα σε μικρότερα υποπροβλήματα, που έχουν την ίδια τυποποίηση με το αρχικό πρόβλημα, αλλά είναι μικρότερα σε μέγεθος. Με όμοιο τρόπο, τα υποπροβλήματα αυτά μπορούν να διαιρεθούν σε ακόμη μικρότερα υποπροβλήματα κ.ο.κ. Έτσι η επίλυση ενός προβλήματος έγκειται στη σταδιακή επίλυση των όσο το δυνατόν μικρότερων υποπροβλημάτων, ώστε τελικά να προκύψει η συνολική λύση του αρχικού ευρύτερου προβλήματος.

Η προσέγγιση αυτή ονομάζεται «από πάνω προς τα κάτω» (top-down).

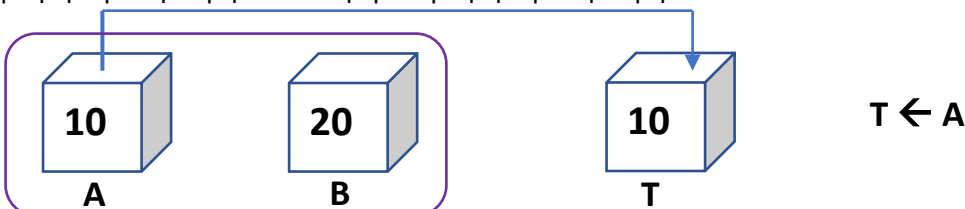
7. 4^η Λειτουργία - Ταξινόμηση ενός στοιχείου του πίνακα.

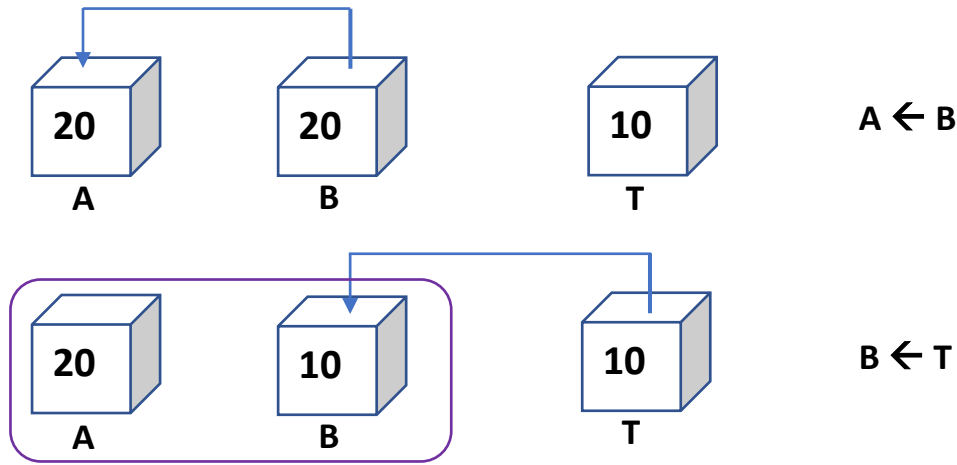
Ταξινόμηση ονομάζουμε την τακτοποίηση των κόμβων μιας δομής με μια ιδιαίτερη σειρά. Η ταξινόμηση μπορεί να γίνει: κατ' αύξουσα σειρά ή κατά φθίνουσα σειρά. Αν και οι αλγόριθμοι ταξινόμησης που θα παρουσιάσουμε είναι απλοί σε υλοποίηση δεν είναι αποτελεσματικοί όσο η **quick sort** την οποία όμως δεν θα την παρουσιάσουμε.

Bubblesort – ταξινόμηση φυσαλίδας ή ευθείας ανταλλαγής (Οδηγός μελέτης 4.3.1)

```
Για I από 2 μέχρι N
  Για J από N μέχρι I με βήμα -1
    Αν B[J] > B[J-1] τότε
      T <-- B[J]
      B[J] <-- B[J-1]
      B[J-1] <-- T
    Τέλος_αν
  Τέλος_επανάληψης
Τέλος_επανάληψης
```

Η αντιμετάθεση των τιμών δύο μεταβλητών πραγματοποιείται με 3 εντολές και την χρήση μιας βοηθητικής μεταβλητής. Ας δούμε αναλυτικά τι κάνει η κάθε μια εντολή στο παρακάτω παράδειγμα. Έστω ότι έχουμε την μεταβλητή **A** με την τιμή 10 και την μεταβλητή **B** με την τιμή 20:





Προσοχή

Δεν υπάρχει συνάρτηση ή διαδικασία με το όνομα **Αντιμετάθεση()**. Συχνά γίνεται η χρήση της σε αλγόριθμους, για μια πιο σύντομη γραφή, όμως απαγορεύεται αυστηρά σε προγράμματα!

Ταξινόμηση αν έχουμε παράλληλους πίνακες

Παράδειγμα 1

Γράψε ένα αλγόριθμο που αποθηκεύει σε δύο πίνακες τα ονόματα και τους τελικούς βαθμούς 100 μαθητών και στην συνέχεια εμφανίζει τους μαθητές ταξινομημένους με φθίνουσα σειρά με βάση τον βαθμό τους.

Πρέπει να ταξινομήσουμε τον πίνακα **B** όμως όταν έχουμε παράλληλους πίνακες πρέπει να διατηρούμε την συσχέτισή τους. Όποτε κατά την διάρκεια της ταξινόμησης όταν μετακινούμε τις τιμές σε δύο στοιχεία στον πίνακα **B** θα πρέπει ταυτόχρονα να αντιμεταθέτουμε και τις αντίστοιχες τιμές στο πίνακα **ON**.



Άρα **αν** λόγω της ταξινόμησης πρέπει να αντιμεταθέσω τις τιμές του **B[1] <- -> B[4]** θα κάνω ακριβώς το ίδιο και στα αντίστοιχα στοιχεία του πίνακα **ON[1] <- -> ON[4]**.

```

Για I από 2 μέχρι N
  Για J από N μέχρι I με βήμα -1
    Αν B[J] > B[J-1] τότε
      T <-- B[J]
      B[J] <-- B[J-1]
      B[J-1] <-- T
      T1 <-- ON[J-1]
      ON[J] <-- ON[J-1]
      ON[J-1] <-- T1
    ΤΕΛΟΣ_ΑΝ
  Τέλος_επανάληψης
Τέλος_επανάληψης
    
```

B[J] <- -> B[J-1]

ON[J] <- -> ON[J-1]

Διπλή ταξινόμηση με παράλληλους πίνακες

Παράδειγμα 2

Να ταξινομήσετε N μαθητών με βάση τον βαθμό τους (πίνακας B[]), σε περίπτωση που έχουν ίδιο βαθμό να ταξινομούνται με βάση το όνομα (πίνακας ON[]).

```

Για I από 2 μέχρι N
  Για J από N μέχρι I με βήμα -1
    Αν B[J] > B[J-1] τότε
      T <-- B[J]
      B[J] <-- B[J-1]
      B[J-1] <-- T
      T1 <-- ON[J-1]
      ON[J] <-- ON[J-1]
      ON[J-1] <-- T1
    ΑΛΛΙΩΣ_ΑΝ B[J]=B[J-1] ΚΑΙ ON[J]<ON[J-1] ΤΟΤΕ
      T1 <-- ON[J-1]
      ON[J] <-- ON[J-1]
      ON[J-1] <-- T1
  ΤΕΛΟΣ_ΑΝ
Τέλος_επανάληψης
Τέλος_επανάληψης
    
```

SELECTION SORT – Ταξινόμηση με επιλογή

Η ταξινόμηση με επιλογή (selection sort), υλοποιείται σε έναν 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

pos ← i

min ← table[i]

Για j από i + 1 μέχρι n

Αν table[j] < min τότε

pos ← j

min ← table[j]

Τέλος_αν

Τέλος_επανάληψης

table[pos] ← table[i]

table[i] ← min

Τέλος_επανάληψης

Τέλος Selection_Sort

Παράδειγμα: Αν υποθέσουμε ότι θέλουμε να ταξινομήσουμε τον παρακάτω πίνακα A[8]

46	55	12	42	94	18	6	67
----	----	----	----	----	----	---	----

Βήμα 1: Εύρεση του ελάχιστου των στοιχείων (7^η θέση) και ανταλλαγή με το στοιχείο της 1^{ης} θέσης

1o	2o	3o	4o	5o	6o	7o	8o
46	55	12	42	94	18	6	67

Βήμα 2: Επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το δεύτερο στοιχείο και έπειτα, το μικρότερο στοιχείο είναι στην 3^η θέση και κάνω ανταλλαγή με το στοιχείο της 2^{ης} θέσης.

1o	2o	3o	4o	5o	6o	7o	8o
6	55	12	42	94	18	46	67

Βήμα 3: Επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το τρίτο στοιχείο και έπειτα, το μικρότερο στοιχείο είναι στην 6^η θέση και κάνω ανταλλαγή με το στοιχείο της 3^{ης} θέσης.

1ο	2ο	3ο	4ο	5ο	6ο	7ο	8ο
6	12	55	42	94	18	46	67

Βήμα 4: Επανάληψη στο τμήμα του πίνακα από το τέταρτο στοιχείο και έπειτα, το μικρότερο στοιχείο είναι στην 4^η θέση και κάνω ανταλλαγή με το στοιχείο της 4^{ης} θέσης. (Αρα παραμένει στην ίδια θέση)

1ο	2ο	3ο	4ο	5ο	6ο	7ο	8ο
6	12	18	42	94	55	46	67

Βήμα 5: Επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το πέμπτο στοιχείο και έπειτα, το μικρότερο στοιχείο είναι στην 7^η θέση και κάνω ανταλλαγή με το στοιχείο της 5^{ης} θέσης.

1ο	2ο	3ο	4ο	5ο	6ο	7ο	8ο
6	12	18	42	94	55	46	67

Βήμα 6: Επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το έκτο στοιχείο και έπειτα, το μικρότερο στοιχείο είναι στην 6^η θέση και κάνω ανταλλαγή με το στοιχείο της 6^{ης} θέσης.

1ο	2ο	3ο	4ο	5ο	6ο	7ο	8ο
6	12	18	42	46	55	94	67

Βήμα 7: Επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το έβδομο στοιχείο και έπειτα, το μικρότερο στοιχείο είναι στην 8^η θέση και κάνω ανταλλαγή με το στοιχείο της 7^{ης} θέσης.

1ο	2ο	3ο	4ο	5ο	6ο	7ο	8ο
6	12	18	42	46	55	94	67

Δε χρειάζεται 8η επανάληψη αφού αν έχω ταξινομήσει τα πρώτα 7 στοιχεία το τελευταίο είναι στην σωστή θέση του!!

8. 5^η Λειτουργία – Συγχώνευση (Οδηγός μελέτης 4.3.3)

Σκοπός της συγχώνευσης είναι η δημιουργία ενός νέου πίνακα που θα περιέχει τα δεδομένα δύο οι περισσότερων άλλων πινάκων.

Παράδειγμα 1

Έστω ότι έχουμε τους πίνακες :

	1	2	3	4		1	2	3	4	5
A	12	-4	9	2	B	32	4	19	-9	3

και θέλουμε να τους συγχωνεύσουμε σε ένα τρίτο πίνακα Γ

	1	2	3	4	5	6	7	8	9
Γ									

Το μέγεθος του πίνακα Γ θα πρέπει να είναι όσο το άθροισμα του πλήθους των στοιχείων του Α και Β δηλαδή σε αυτό το παράδειγμα να έχει 9 θέσεις. Προσοχή, οι πίνακες Α, Β και Γ πρέπει να είναι του ίδιου τύπου!

Βήμα 1: Επιλέγουμε τυχαία ένα από τους δύο πίνακες και αντιγράφουμε ένα-ένα τα στοιχεία του στον νέο πίνακα Γ

	1	2	3	4		1	2	3	4	5	6	7	8	9
A	12	-4	9	2										
Γ	12	-4	9	2										

ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 4
 $\Gamma[I] \leftarrow A[I]$
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Βήμα 2: Στην συνέχεια κάνουμε το ίδιο για τον δεύτερο πίνακα, όμως προσέχουμε να τοποθετήσουμε τα στοιχεία του στις κενές θέσεις του πίνακα Γ που είναι από την 5^η και μετά.

	1	2	3	4	5		1	2	3	4	5
B	32	4	19	-9	3						
Γ	12	-4	9	2	32	4	19	-9	3		

ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 5
 $\Gamma[4+I] \leftarrow B[I]$
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Συγχώνευση δύο ταξινομημένων πινάκων.

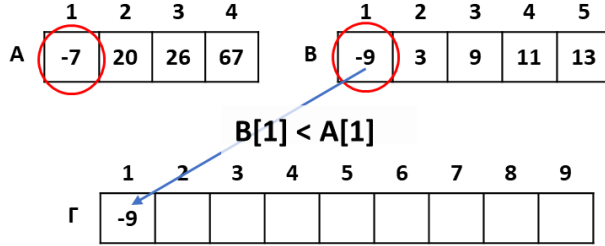
Έχουμε δύο ταξινομημένους πίνακες

	1	2	3	4
A	-7	20	26	67

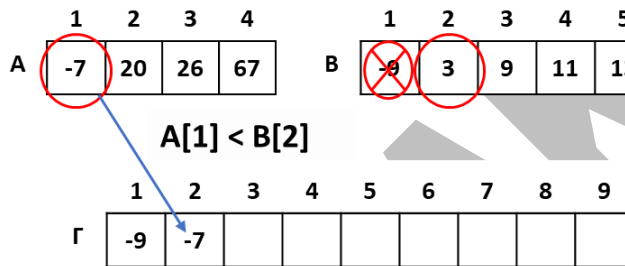
	1	2	3	4	5
B	-9	3	9	11	13

και θέλουμε να δημιουργήσουμε ένα νέο ταξινομημένο πίνακα Γ[] που να περιέχει τα στοιχεία τους:

Βήμα 1: Αρχικά εντοπίζουμε την μικρότερη τιμή και στους δύο πίνακες. Γνωρίζοντας ότι οι πίνακες είναι ταξινομημένοι αυτή η τιμή θα βρίσκεται είτε στο A[1] ή στο B[1].



Με τον ίδιο τρόπο συνεχίζουμε για να βρούμε το δεύτερο μικρότερο στοιχείο το οποίο θα είναι είτε στο A[1] ή στο B[2]



Η παραπάνω διαδικασία κατά την οποία συγκρίνουμε ένα στοιχείο του πίνακα A με ένα στοιχείο του πίνακα B συνεχίζει μέχρι ένας από τους πίνακες να αδειάσει.

```

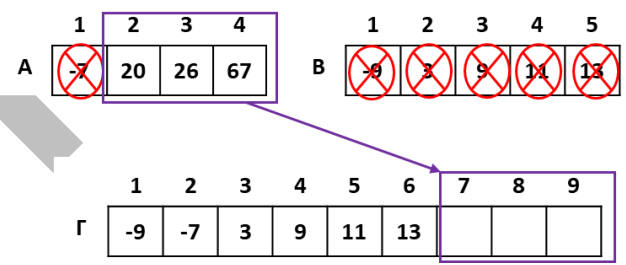
αi <-- 1
βi <-- 1
γi <-- 1
ΟΣΟ αi <= 4 ΚΑΙ βi <= 5 ΕΠΑΝΑΛΑΒΕ
  ΑΝ A[αi] < B[βi] ΤΟΤΕ
    Γ[γi] <-- A[αi]
    αi <-- αi + 1
  ΑΛΛΙΩΣ
    Γ[γi] <-- B[βi]
    βi <-- βi + 1
  ΤΕΛΟΣ_ΑΝ
  γi <-- γi + 1
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    
```

Όταν τοποθετώ ένα στοιχείο του A θα πρέπει να αυξάνω και τον δείκτη του αi+1 ώστε να δείχνει στην επόμενη θέση!

Όταν τοποθετώ ένα στοιχείο του B θα πρέπει να αυξάνω και τον δείκτη του βi+1 ώστε να δείχνει στην επόμενη θέση!

Σε κάθε επανάληψη στον πίνακα Γ θα τοποθετώ και ένα νέο στοιχείο οπότε δεν ξερνώ να μετακινήσω τον δείκτη του ώστε να δείχνει στην επόμενη άδεια θέση!

Βήμα 2 : Κάποια στιγμή ένας από τους δύο πίνακες θα αδειάσει, στο παράδειγμα μας αυτός θα είναι ο πίνακας A.



```

ΑΝ αi <= 4 ΤΟΤΕ
  ΓΙΑ I ΑΠΟ αi ΜΕΧΡΙ 4
    Γ[γi] <-- A[I]
    γi <-- γi + 1
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΛΛΙΩΣ
  ΓΙΑ I ΑΠΟ βi ΜΕΧΡΙ 5
    Γ[γi] <-- B[I]
    γi <-- γi + 1
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΑΝ
    
```

Ελέγχοντας τους δείκτες αi και βi εντοπίζουμε ποιος πίνακας έχει ακόμα στοιχεία και αντιγράφουμε ένα-ένα τα στοιχεία του στις τελευταίες θέσεις του νέου πίνακα.

9. Άλλες εργασίες σε πίνακες

Διαχωρισμός ενός πίνακα

Κατά τον διαχωρισμό χωρίζουμε τα στοιχεία ενός πίνακα σε δύο οι περισσότερους πίνακες με βάση κάποια συνθήκη.

Παράδειγμα

Ας υποθέσουμε ότι θέλουμε να χωρίσουμε τα στοιχεία του πίνακα Γ σε θετικούς και μη θετικούς αριθμούς δημιουργώντας δύο άλλους πίνακες τον A που θα περιέχει τους μη θετικούς αριθμούς και τον Θ για τους θετικούς. Το μέγεθος του κάθε νέου πίνακα θα πρέπει να είναι όσο και του αρχικού αφού θα μπορούσαν όλα τα στοιχεία να είναι θετικοί αριθμοί ή αρνητικοί αριθμοί!

Έστω ο πίνακας

	1	2	3	4	5	6	7	8	9
Γ	5	20	-3	8	31	-2	45	-9	10

Άρα θέλουμε να χωρίσουμε τα στοιχεία του πίνακα και να δημιουργήσουμε τους νέους πίνακες ως εξής:

	1	2	3	4	5	6	7	8	9
Θ	5	20	8	31	45	10			

	1	2	3	4	5	6	7	8	9
A	-3	-2	-9						

Προσοχή συχνό λάθος είναι η δημιουργία των νέων πινάκων ως:

	1	2	3	4	5	6	7	8	9
Θ	5	20		8	31		45		10



	1	2	3	4	5	6	7	8	9
A			-3				-2		-9

Δεν πρέπει να αφήνουμε κενές θέσεις ανάμεσα στις τιμές που τοποθετούμε στους πίνακες μας!!

Ο αλγόριθμος του διαχωρισμού απαιτεί την χρήση διαφορετικής μεταβλητής ως δείκτη για τους τρεις πίνακες.

```

 $\theta_i$  <-- 1
 $\alpha_i$  <-- 1
Για  $I$  από 1 μέχρι 9
  Αν  $\Gamma[I] \leq 0$  τότε
     $A[\alpha_i]$  <--  $\Gamma[I]$ 
     $\alpha_i$  <--  $\alpha_i + 1$ 
  Αλλιώς
     $\Theta[\theta_i]$  <--  $\Gamma[I]$ 
     $\theta_i$  <--  $\theta_i + 1$ 
Τέλος_αν
Τέλος_επανάληψης
    
```

I : Ο δείκτης του αρχικού πίνακα Γ] σε κάθε επανάληψη αυξάνετε κατά ένα ώστε να ελέγγω σταδιακά ένα-ένα τα στοιχεία του.

θ_i : Ο δείκτης του πίνακα Θ] αυξάνετε μόνο αν βρω και τοποθετήσω ένα θετικό αριθμό στον πίνακα Θ .

α_i : Ο δείκτης του πίνακα A] αυξάνετε μόνο αν βρω και τοποθετήσω ένα μη θετικό αριθμό στον πίνακα A .

Αντιστροφή των στοιχείων ενός πίνακα χωρίς την χρήση δεύτερου πίνακα.

Παράδειγμα

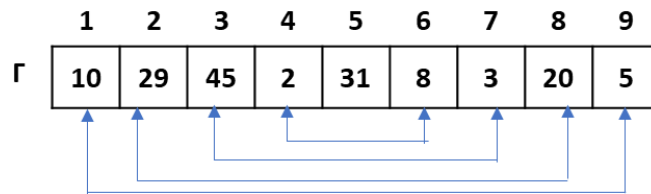
Έστω ότι έχουμε τον πίνακα :

	1	2	3	4	5	6	7	8	9
Γ	5	20	3	8	31	2	45	9	10

Θέλουμε να αντιστρέψουμε τα στοιχεία του πίνακα, δηλαδή το τελευταίο να πάει στην 1^η θέση, το προτελευταίο στην 2^η θέση κ.τ.λ. Έτσι ώστε στο τέλος ο πίνακας μας να έχει την μορφή που βλέπουμε παρακάτω.

	1	2	3	4	5	6	7	8	9
Γ	10	29	45	2	31	8	3	20	5

Ήδη από την ταξινόμηση γνωρίζουμε πολύ καλά να αντιμετωπίσουμε τις τιμές 2 μεταβλητών. Όπως παρατηρούμε στην παρακάτω εικόνα πρέπει να αντιμετωπίσουμε τις τιμές σε 4 ζευγάρια (**Μέγεθος div 2**).



```

ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 4
  t <-- Γ[I]
  Γ[I] <-- Γ[10-I]
  Γ[10-I] <-- t
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    
```

```

Γ[I] <--> Γ[10-I]
Γ[1] <--> Γ[9]  όταν το I είναι 1
Γ[2] <--> Γ[8]  όταν το I είναι 2
Γ[3] <--> Γ[7]  όταν το I είναι 3
Γ[4] <--> Γ[6]  όταν το I είναι 4
    
```

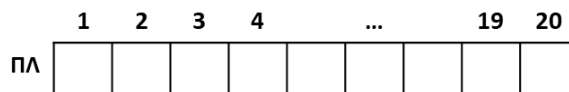
Δημιουργία νέου πίνακα (πλήθη ή Αθροίσματα)

Παράδειγμα (Οδηγός μελέτης ασκ. 13 σελ. 95) Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο σε ένα μονοδιάστατο πίνακα ΒΑΘΜΟΙ[52] καταχωρεί τους βαθμούς του μαθήματος Πληροφορικής, των 52 μαθητών της Γ' Λυκείου ενός σχολείου. Οι βαθμοί θεωρούνται θετικοί και ακέραιοι. Το πρόγραμμα να υπολογίζει και να τυπώνει τη συχνότητα που εμφανίζεται ο κάθε βαθμός, αν θεωρήσουμε ότι όλοι οι βαθμοί είναι από το 1 μέχρι το 20.

Για να βρούμε την συχνότητα κάθε βαθμού θα πρέπει να γνωρίζουμε πόσοι μαθητές έχουν πάρει τον συγκεκριμένο βαθμό. Άρα χρειαζόμαστε 20 διαφορετικές μεταβλητές που θα χρησιμοποιηθούν ως πλήθος. π.χ.

ΠΛ1 ← 0 θα μετράει πόσοι μαθητές πήραν βαθμό 1 , **ΠΛ2** ← 0 θα μετράει πόσοι πήραν βαθμό 2 κ.τ.λ.

Όμως δεν είναι πρακτικό να δηλώσουμε τόσες πολλές διαφορετικές μεταβλητές θα πρέπει να χρησιμοποιήσουμε ένα πίνακα ο οποίος θα έχει 20 θέσεις.



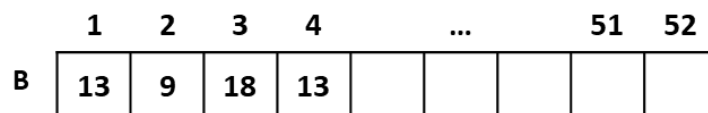
Μέσα σε κάθε θέση θα υπάρχει ένα νούμερο το οποίο θα δείχνει πόσες φορές εμφανίστηκε ο βαθμός που αντιστοιχεί στην συγκεκριμένη θέση. Σε αυτή την άσκηση η αντιστοίχιση είναι εύκολη αφού στην πρώτη θέση ΠΛ[1] θα μετράμε των πλήθος αυτών που πήρανε βαθμό 1, στην θέση ΠΛ[2] τι πλήθος αυτών που πήρανε βαθμό 2 κ.τ.λ.

Προσοχή: Δεν ξεχνάμε, όπως κάθε μεταβλητή που μετράει πλήθος έτσι και εδώ θα πρέπει αρχικά να μηδενίσουμε κάθε θέση του πίνακα!

```

ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 20
  ΠΛ[I] <-- 0
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    
```

Στην συνέχεια ελέγχουμε ένα – ένα τα στοιχεία του πίνακα Β για να δούμε τον βαθμό του κάθε μαθητή.



Έστω ότι ο βαθμός του πρώτου μαθητή (Β[1]) είναι 13. Άρα θα πρέπει να αυξήσουμε την τιμή της 13^{ης} θέσης του πίνακα ΠΛ κατά ένα. Ο Βαθμός δηλαδή δείχνει ποια θέση του πίνακα ΠΛ πρέπει να αυξηθεί.

```

ΒΑΘ <-- Β[I]
ΠΛ[ΒΑΘ] <-- ΠΛ[ΒΑΘ] + 1
    
```

Την παραπάνω διαδικασία την κάνω για κάθε ένα στοιχείο (μαθητή) του πίνακα ΒΑΘ

```
ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 52
```

```
ΒΑΘ <-- Β[I]
```

```
ΠΛ[ΒΑΘ] <-- ΠΛ[ΒΑΘ] + 1
```

```
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

```
ΠΛ[B[I]] <-- ΠΛ[B[I]] + 1
```

Στο τέλος εμφανίζουμε τη συχνότητα του κάθε βαθμού που είναι το ποσοστό εμφάνισης του βαθμού στο σύνολο των 52 μαθητών.

```
ΓΙΑ I ΑΠΟ 1 ΜΕΧΡΙ 20
```

```
ΓΡΑΨΕ ΠΛ[I]/52 * 100
```

```
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```