

# Εισαγωγή στη γλώσσα προγραμματισμού C++

Πίνακες

Άριστος Πασιάς

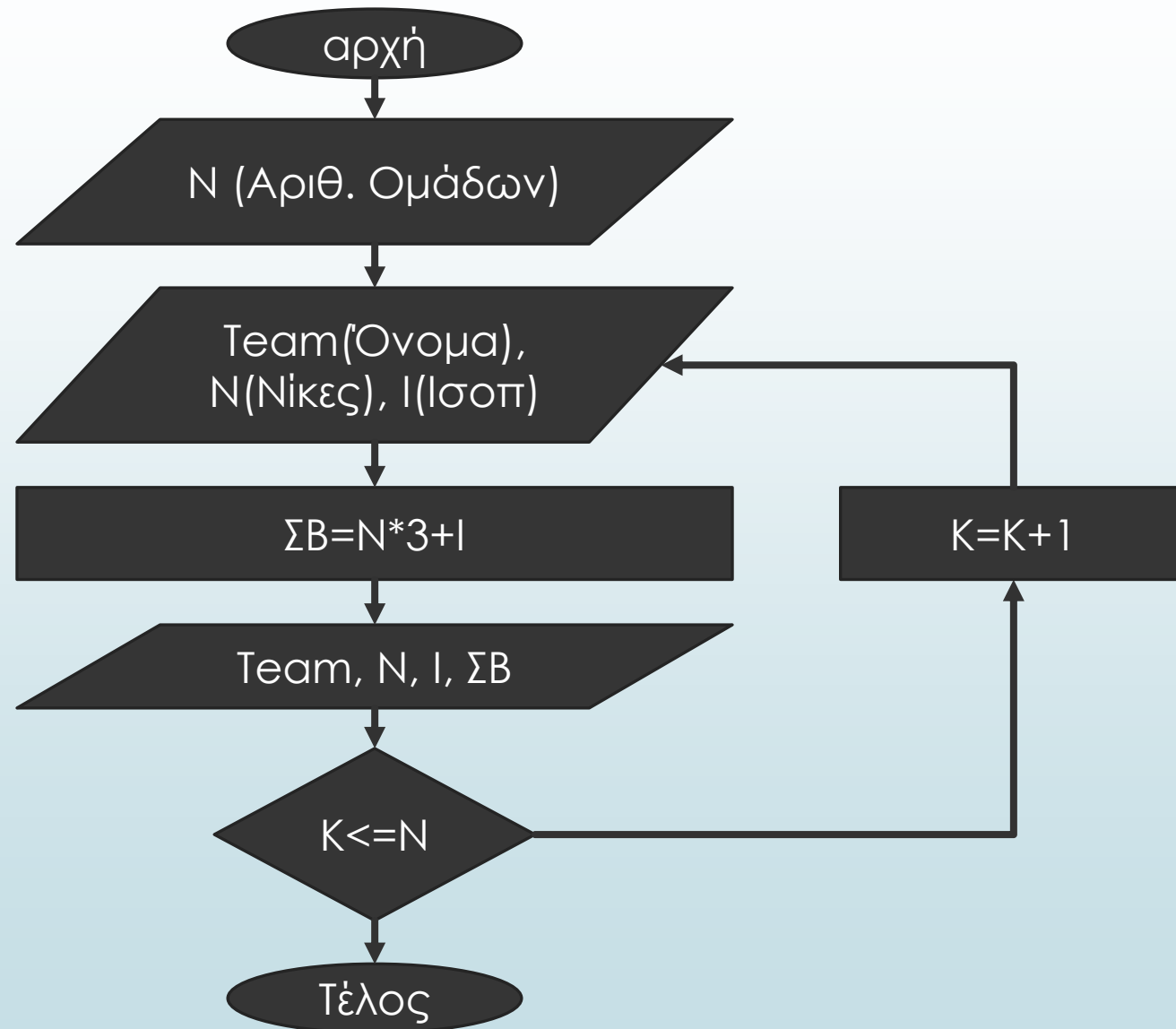
# Επίλυση προβλήματος

Να δημιουργήσετε ένα πρόγραμμα το οποίο να δέχεται το όνομα, κάθε ομάδας ποδοσφαίρου Α' κατηγορίας, τον αριθμό των νικών και ισοπαλιών που έχει πετύχει και :

- α) Να υπολογίζει και να εμφανίζει το σύνολο των βαθμών που συνέλεξε κάθε ομάδα αν για κάθε νίκη κερδίζει 3 βαθμούς και για κάθε ισοπαλία 1 βαθμό.
- β) Να κατατάξετε τις ομάδες με την σειρά ώστε πρώτη να εμφανίζεται η ομάδα με τους περισσότερους βαθμούς και τελευταία η ομάδα με τους λιγότερους βαθμούς

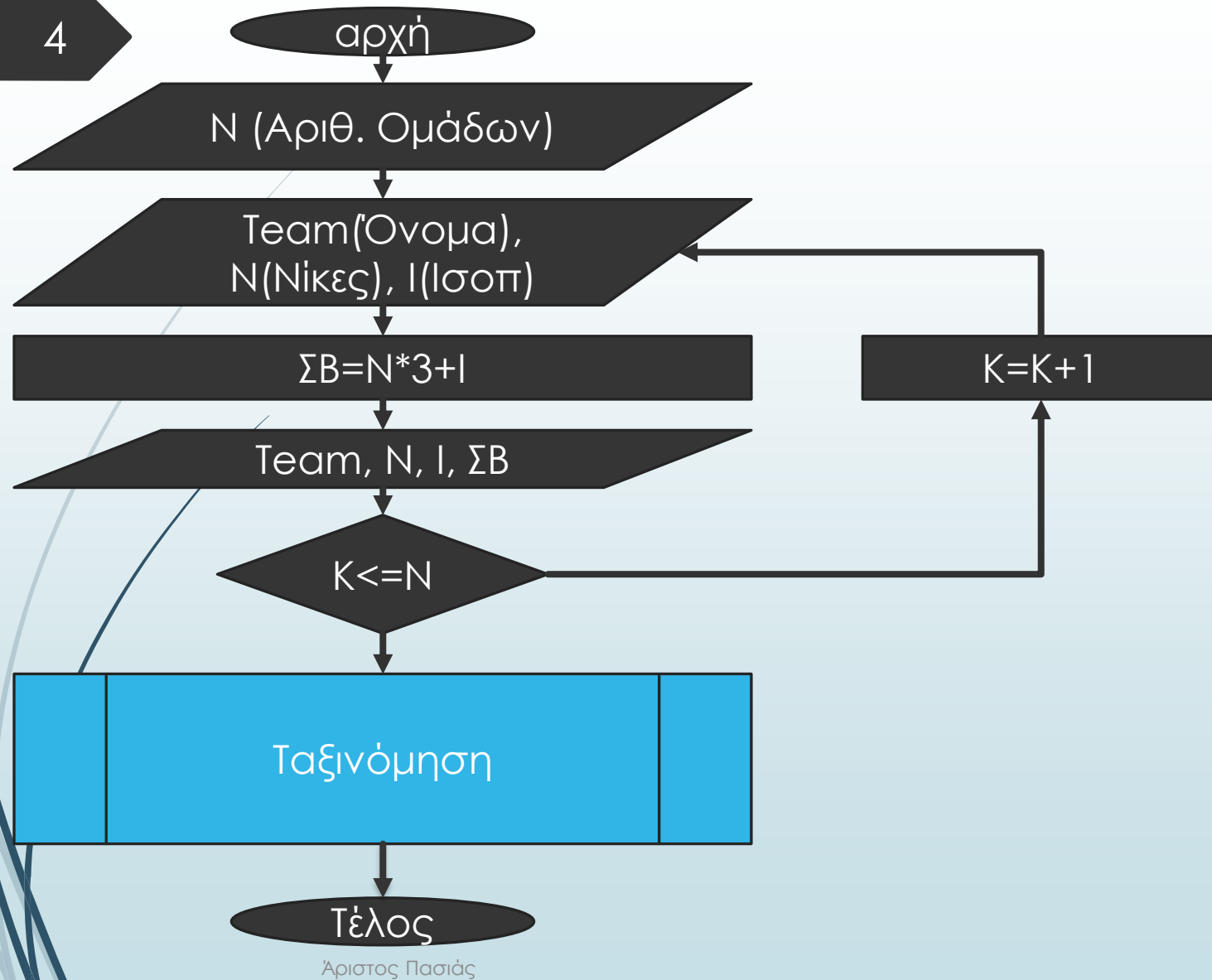
# Επίλυση προβλήματος (α)

3



# Επίλυση προβλήματος (β)

4



Άριστος Πασιάς

Για να μπορέσω να δημιουργήσω και να εμφανίσω την βαθμολογία χρειάζεται να αποθηκεύω τα αποτελέσματα κάθε βρόχου σε μεταβλητές προκειμένου να μπορέσω αργότερα να ταξινομήσω τα αποτελέσματα μου. Τι γίνεται όμως αν μιλάμε για χιλιάδες στοιχεία; Πόσες μεταβλητές θα χρειαστώ;

# 1. Πίνακες (Arrays) Εισαγωγή

Στη C++, όπως και σε όλες τις γλώσσες προγραμματισμού, υπάρχει η δυνατότητα ομαδοποίησης δεδομένων **ίδιου τύπου**.

Ο τρόπος με τον οποίο επιτυγχάνεται αυτό είναι με τη δημιουργία πινάκων (arrays).

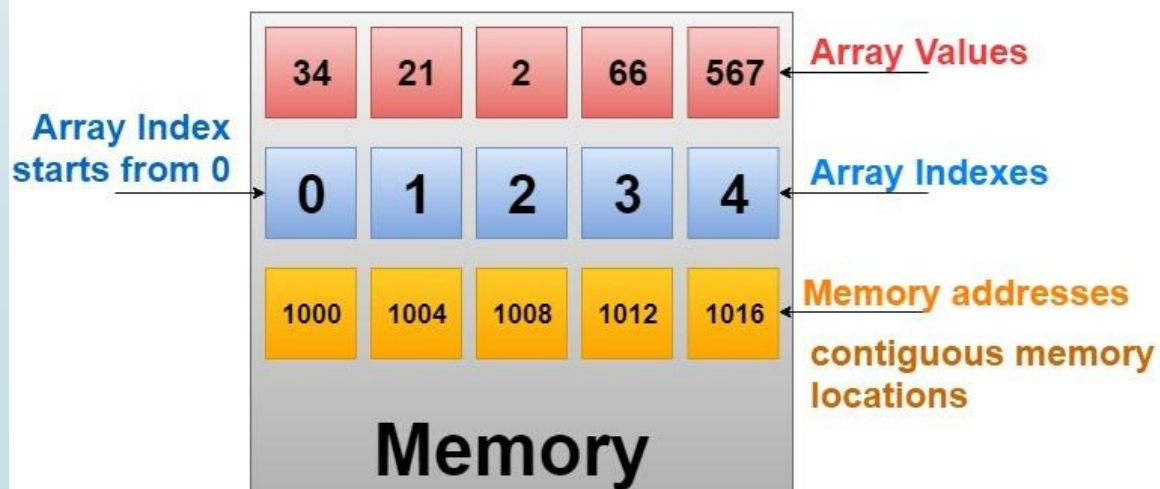
Αναφορά στον πίνακα γίνεται μέσω ενός ονόματος όπως στις μεταβλητές, ενώ για πρόσβαση στα δεδομένα ενός πίνακα χρησιμοποιείται το όνομα και ένας αριθμός (δείκτης - index), ο οποίος καθορίζει τη θέση του στοιχείου μέσα στον πίνακα.

Τα πέντε στοιχεία του πίνακα στο δεξιό σχήμα αναφέρονται ως:

**x[0], x[1], x[2], x[3], x[4]**

Οι αριθμοί στις αγκύλες ονομάζονται **Δείκτες**

```
int x[5] = {34, 21, 2, 66, 567}
```



# Ορισμός πίνακα

- Ουσιαστικά ένας πίνακας είναι μια μεταβλητή που μπορεί να κρατήσει πολλές τιμές του ίδιου τύπου. Χρησιμοποιούμε ένα δείκτη για να αναφερθούμε στην τιμή που μας ενδιαφέρει κάθε φορά.
- Πρέπει λοιπόν να ορίσουμε τον πίνακα και το πλήθος των τιμών που περιλαμβάνει πριν από την χρήση του όπως ακριβώς κάνουμε και με τις μεταβλητές.
- Π.χ.1 η δήλωση **Int Vathmi[14]** σημαίνει την δήλωση ενός πίνακα που θα περιέχει 14 τιμές από ακαίρους αριθμούς.
- Π.χ.2 η δήλωση **float grades[5] = {12.5, 14.0, 18.5, 20.0, 19.2}** σημαίνει την δήλωση ενός πίνακα που θα περιέχει 5 πραγματικούς αριθμούς οι τιμές των οποίων ορίζονται άμεσα με την δήλωση του πίνακα.
- **string cars[4] = {"Volvo", "BMW", "Ford", "Mazda"};**

# Ορισμός πίνακα

7

- Η δήλωση του μεγέθους μέσα στις αγκύλες **δεν είναι απαραίτητη** όταν αποδίδουμε αρχικές τιμές, αφού ο μεταγλωττιστής μπορεί να καταμετρήσει τις τιμές μέσα στις παρενθέσεις.
- Αν δώσουμε το μέγεθος του πίνακα και το πλήθος των αρχικών τιμών μέσα στα άγκιστρα είναι μικρότερο, δεν θα υπάρχει πρόβλημα, καθώς στα στοιχεία του πίνακα που απομένουν θα αποδοθεί ως αρχική τιμή το 0.
- Αν όμως το πλήθος των αρχικών τιμών μέσα στα άγκιστρα είναι μεγαλύτερο από το μέγεθος που δηλώθηκε, τότε υπάρχει σφάλμα.
- `int primes[] = {1,2,3,5,7};` // Δεν χρειάζεται να δηλώνουμε το μέγεθος.
- `char characters[3] = {'a', 'b', 'c', 'd'};` // **Σφάλμα!**  
**Περισσότερα στοιχεία.**
- `int a[5] = {12,5,4};` // `a` → `{12, 5, 4, 0, 0}`

# Αναφορά σε στοιχεία πίνακα

8

- Όταν θέλουμε να αναφερθούμε σε κάποιο στοιχείο του πίνακα απλά χρησιμοποιούμε το όνομα του πίνακα και τον αριθμό του αντίστοιχου δείκτη.
- Προσοχή το πρώτο στοιχείο έχει δείκτη «0»
  - `Int arr[3];` //Δήλωση πίνακα 3 ακέραιων τιμών
  - `arr[0] = 5;` //Το πρώτο στοιχείο παίρνει τιμή 5
  - `cin >> arr[1];` //Εισαγωγή τιμής του στοιχείου 2
  - `arr[2] = arr[0] + arr[1] + 1;` //Αριθμητικές Πράξεις
  - `Cout<<arr[2]` //Εμφάνιση στοιχείου πίνακα

# Εισαγωγή & Εμφάνιση στοιχείων πίνακα

9

► Για την εισαγωγή και την εμφάνιση στοιχείων σε πίνακα μπορούμε να χρησιμοποιήσουμε τη δομή επανάληψης **for** μαζί με μία μεταβλητή-δείκτη.

- `char arr[10]; // Ορισμός πίνακα`
- `for (int i=0; i<10; i++)`
- `cin >> arr[i]; // Εισαγωγή στοιχείων πίνακα`
- `for (int i=0; i<10; i++)`
- `cout << arr[i] << " "; // Εμφάνιση  
στοιχείων πίνακα`

# Παράδειγμα1 (pxt1)

10

- Να δημιουργήσετε πρόγραμμα το οποίο να αποθηκεύει τους χαρακτήρες 'O', 'L', 'L', 'E', 'H' σε έναν πίνακα χαρακτήρων και να τους εμφανίζει στην οθόνη αντίστροφα, δηλαδή ο πρώτος να εμφανίζεται τελευταίος, ο δεύτερος να εμφανίζεται προτελευταίος.

```
1. #include<iostream>
2. using namespace std;
3. int main() {
4. char arr[] = {'O', 'L', 'L', 'E', 'H'};
5. for (int i = 4; i >= 0; i--)
6. cout << arr[i] << " ";
7. return 0;
8. }
```

# Παράδειγμα 2 (pxt2)

11

- Να δημιουργήσετε πρόγραμμα το οποίο να διαβάζει δέκα ακέραιους αριθμούς, να τους αποθηκεύει σε πίνακα και να επιστρέφει το πλήθος των στοιχείων του πίνακα, τα οποία είναι μεγαλύτερα από τον μέσο όρο όλων των στοιχείων του πίνακα.

```
1. #include<iostream>
2. using namespace std;
3. int main(){
4.     int arr[10], sum = 0, ans = 0,i;
5.     float avg;
6.     for (i = 0; i < 10; i++){
7.         cin >> arr[i];
8.         sum += arr[i]; // αντιστοιχεί με sum = sum + arr[i]
9.     }
10.    avg = float(sum) / 10;
11.    for (i = 0; i < 10; i++)
12.        {if (arr[i]>avg)
13.            ans++;
14.        }
15.    cout << ans;
16.    return 0;
17.}
```

# Παράδειγμα 3 (ext9)

12

- Να δημιουργήσετε πρόγραμμα το οποίο να γεμίζει έναν πίνακα ακεραίων 30 θέσεων με τους 30 πρώτους αριθμούς της ακολουθίας: 1, 3, 7, 15, 31, 63,  $2_{N-1}+1$ . Ακολούθως, να ζητά από τον χρήστη έναν αριθμό N ( $1 \leq N \leq 30$ ) και να επιστρέφει τον N-1, τον N-οστό και τον N+1 αριθμό της πιο πάνω σειράς.

```
1. #include<iostream>
2. using namespace std;
3. int main(){
4.     int arr[30], N;
5.     arr[0]=1; // Γνωρίζουμε το πρώτο στοιχείο
6.     for (int i = 1; i < 30; i++){
7.         arr[i] = 2 * arr[i-1] + 1;} // Από το προηγούμενο στοιχείο βρίσκουμε το επόμενο
8.     cin >> N; // Εισαγωγή θέσης στοιχείου
9.     cout << arr[N-1]<< arr[N]<< arr[N+1];
10. return 0;
11. }
```

# Παράλληλοι πίνακες

13

- Πολλές φορές έχουμε περισσότερα δεδομένα, διαφορετικών τύπων, για να χειριστούμε. Για παράδειγμα, έχουμε τον πιο κάτω πίνακα, ο οποίος παρουσιάζει τα συνολικά κέρδη μίας εταιρείας ανά έτος:

YEAR	PROFIT
2010	97.77
2011	98.34
2012	103.56
2013	102.68
2014	104.54

Για να καταφέρουμε να συσχετίσουμε τα πιο πάνω στοιχεία, μπορούμε να χρησιμοποιήσουμε δύο παράλληλους πίνακες (parallel arrays).

```
int year[] = {2010, 2011, 2012, 2013, 2014};
double profit[] = {97.77, 98.34, 103.56, 102.68, 104.54};
cout << "YEAR" << "\t" << "PROFIT" << endl;
for (int i = 0; i < 5; i++)
    cout << year[i] << "\t" << profit[i] << endl;
```

# Παράδειγμα 4 (ext10)

14

- Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται αρχικά τα στοιχεία μίας παραγγελίας μίας υπεραγοράς. Συγκεκριμένα, το πρόγραμμα θα διαβάζει για 50 διαφορετικά προϊόντα: τον κωδικό του προϊόντος, την ποσότητα και την τιμή και θα τα αποθηκεύει σε κατάλληλους παράλληλους πίνακες. Για κάθε ένα από τα προϊόντα, το πρόγραμμα να υπολογίζει το κόστος (κόστος = ποσότητα x τιμή) και να το αποθηκεύει σε πίνακα και τέλος, να υπολογίζει και να εμφανίζει τη συνολική χρέωση της παραγγελίας. Ενδεικτικά, βλέπετε ένα μέρος της παραγγελίας στον πιο κάτω πίνακα. Οι πρώτες τρεις στήλες παρουσιάζουν τον κωδικό, την ποσότητα και την τιμή. Η τέταρτη στήλη υπολογίζει το κόστος για κάθε προϊόν. Τέλος, εμφανίζεται η συνολική χρέωση της παραγγελίας.

Κωδικός	Ποσότητα	Τιμή	Ποσό
1001	30	3.99	119.70
1002	45	2.49	112.05
1003	12	0.99	11.88
1004	18	5.99	107.82
.	.	.	.
.	.	.	.
.	.	.	.
1050	10	8.49	84.90
<b>Συνολική χρέωση παραγγελίας:</b>			<b>9987.46</b>

# Παράδειγμα 4 Λύση

15

```
1. #include<iostream>
2. using namespace std;
3. const int N=50
4. int main(){
5.     int code[N], quantity[N];
6.     double price[N], total[N], ans = 0;
7.     for (int i = 0; i < N; i++){
8.         cin >> code[i] >> quantity[i] >> price[i];
9.         total[i] = quantity[i] * price[i];
10.        ans += total[i];
11.    }
12.    cout << ans;
13.    return 0;
14. }
```

Απόστολος Πισιδάς



# Ασκήσεις

16

## Άσκηση 1 ext11

Να γράψετε μια συνάρτηση σε C++ που να αντιγράφει τα στοιχεία ενός πίνακα  $A[ ]$  μεγέθους  $N$  στον πίνακα  $B[ ]$ .

## Άσκηση 2 ext12

Έχουμε δυο πίνακες, τον  $X[ ]$  και τον  $Y[ ]$  με μέγεθος  $N$ . Να δημιουργήσετε πρόγραμμα το οποίο να αντιμεταθέτει τις τιμές των δυο πινάκων.

## Άσκηση ext13

Θεωρήστε ένα πίνακα μεγέθους  $N$  συμπληρωμένο με τυχαίους θετικούς ακεραίους τετραψήφιους αριθμούς. Να γράψετε πρόγραμμα σε C++ που ο χρήστης θα δίνει ένα αριθμό και το πρόγραμμα θα ελέγχει αν υπάρχει στον πίνακα και σε ποια θέση.

## Άσκηση ext14

Να γράψετε μια συνάρτηση σε C++ η οποία θα αντιστρέφει τα στοιχεία ενός πίνακα μεγέθους  $N$ . (Το πρώτο στοιχείο να μεταφερθεί στο τέλος, το τελευταίο πρώτο κ.ο.κ.)

## Άσκηση ext15

Θεωρήστε ένα πίνακα  $A[ ]$  με 20 στοιχεία και τυχαίες τιμές από το 1 ως το 100. Γράψτε ένα πρόγραμμα που θα εντοπίζει τυχόν διπλές εγγραφές.

# Λύσεις Ασκήσεων

17

ext11

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int N, i;
8      cout<<"Insert the array size "<<endl;
9      cin>>N;
10     int A[N],B[N];
11     for(i=0;i<N;i++){
12         cout<<"Insert the " <<i<<" element of the array "<<endl;
13         cin>>A[i];
14     }
15     cout<<"Element \t"<<" Array A \t"<<" Array B "<<endl;
16     for(i=0;i<N;i++){
17         B[i]=A[i];
18         cout<<i<<" \t\t "<<A[i]<<" \t\t "<<B[i]<<endl;
19     }
20     cout << "End of program!" << endl;
21     return 0;
22 }
23
```

ext12

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int A[5]={100,101,102,103,104};
8      int B[5]={1000,1001,1002,1003,1004};
9      int i,temp;
10     cout<<"Before exchange : "<<endl;
11     cout<<"Element \t"<<" Array A \t"<<" Array B "<<endl;
12     for(i=0;i<5;i++){
13         cout<<i<<" \t\t "<<A[i]<<" \t\t "<<B[i]<<endl<<endl;
14     }
15     for(i=0;i<5;i++){
16         temp=A[i];
17         A[i]=B[i];
18         B[i]=temp;
19     }
20     cout<<"After exchange : "<<endl;
21     cout<<"Element \t"<<" Array A \t"<<" Array B "<<endl;
22     for(i=0;i<5;i++){
23         cout<<i<<" \t\t "<<A[i]<<" \t\t "<<B[i]<<endl;
24     }
25     cout << "End of program!" << endl;
26     return 0;
27 }
28
```

# Λύσεις Ασκήσεων

18

ext13

ext14

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int A[5]={1000,1001,1002,1003,1004};
8      int i,numc;
9      bool answer;
10     cout<<"Insert your code "<<endl;
11     cin>>numc;
12     for(i=0;i<5;i++){
13         if(numc==A[i]){
14             answer=true;
15             break;
16         }
17         else answer=false;
18     }
19     if(answer)
20         cout<<"found "<<numc<<" in position "<<i<<endl;
21     else{
22         cout<<numc<<" Do not exist in the array"<<endl;
23     }
24     cout << "End of program!" << endl;
25     return 0;
26 }
27
```

# Ταξινόμηση πίνακα

19

>>Δίνεται πίνακας  $a$  με  $N$  στοιχεία

(επανάληψη 1) Για  $i=0$  έως  $N-2$  με βήμα 1

θέσε  $min=a[i]$

(επανάληψη 2) Για  $j=i+1$  έως  $N-1$  με βήμα 1

**Αν**  $a[j] < min$  **τότε**

θέσε  $min=a[j]$

$swap\ a[j] \leftrightarrow a[i]$

τέλος επανάληψης 2

τέλος επανάληψης 1

Τέλος ταξινόμησης

Μέθοδος 1  
Selection

Μέθοδος 1  
Bubble sort

>>Δίνεται πίνακας  $a$  με  $N$  στοιχεία

(επανάληψη 1) Για  $i=1$  έως  $N-1$  με βήμα 1

θέσε  $Flag=FALSE$

(επανάληψη 2) Για  $j=0$  έως  $N-1-i$  με βήμα 1

**Αν**  $a[j] > a[j+1]$  **τότε**

$swap\ a[j] \leftrightarrow a[j+1]$

θέσε  $Flag=TRUE$

τέλος επανάληψης 2

**Αν**  $Flag=FALSE$  **τέλος ταξινόμησης** (break)

τέλος επανάληψης 1

Τέλος ταξινόμησης

# Ασκήσεις σελίδα 191 βιβλίου

20



# Ασκήσεις σελίδα 191 βιβλίου

21

## Άσκηση 5.12

Στο Παγκύπριο πρωτάθλημα ρυθμικής γυμναστικής λαμβάνουν μέρος 15 αθλήτριες. Η βαθμολογία της κάθε αθλήτριας είναι ένας δεκαδικός αριθμός. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται τα ονόματα και τις βαθμολογίες, να τα αποθηκεύει σε πίνακες και να τυπώνει στην οθόνη το όνομα και την βαθμολογία όσων είναι πάνω από τον μέσο όρο όλων των βαθμολογιών.

## Άσκηση 5.13

Να δημιουργήσετε πρόγραμμα το οποίο να διαβάζει 100 πραγματικούς αριθμούς και να τους αποθηκεύει σε έναν πίνακα. Στη συνέχεια να αναζητά και να εμφανίζει τον μεγαλύτερο αριθμό, καθώς και τη θέση της πρώτης εμφάνισής του στον πίνακα.

## Άσκηση 5.15

Η υπηρεσία περιβάλλοντος καταγράφει την πυκνότητα σκόνης στην ατμόσφαιρα καθημερινά. Η πυκνότητα σκόνης χαρακτηρίζεται από έναν δεκαδικό αριθμό, από το 0.0 μέχρι το 1.0. Να δημιουργήσετε πρόγραμμα το οποίο να:

(α) Καταχωρίζει τα στοιχεία για έναν χρόνο στον πίνακα M.

(β) Υπολογίζει και τυπώνει τον μέσο όρο.

(γ) Βρίσκει και τυπώνει τις μέρες με τη μεγαλύτερη πυκνότητα σκόνης. Σε περίπτωση που υπήρχαν περισσότερες από μία μέρες με μέγιστο επίπεδο σκόνης, να τις τυπώνεις όλες.