Εισαγωγή στον Προγραμματισμό

Ενότητα C++ (75%) :

7η εβδομάδα

Παναγιώτης Τζουνάκης

(συμπεριλαμβάνει υλικό που ευγενικά προσέφερε ο καθηγητής κ. Γεώργιος Ραχώνης)

Χειμώνας 2026

Παράδειγμα

Να γραφεί πρόγραμμα το οποίο, θα διαβάζει *n* αριθμούς

$$x_1, x_2, \ldots, x_n$$

και στη συνέχεια θα υπολογίζει τους τύπους

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n}$$

και

$$S = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \mu)^2}{n-1}}$$

Πίνακας είναι μια σύνθετη μεταβλητή που καταλαμβάνει παραπάνω από μια θέση στην μνήμη του Η/Υ, έχει συγκεκριμένο όνομα και δέχεται ένα συγκεκριμένο τύπο δεδομένων.

Μονοδιάστατοι πίνακες

Δήλωση

΄Τύπος μεταβλητής Όνομα_Πίνακα[πλήθος στοιχείων];

Παράδειγμα

```
int a[10];
float b[30],c[15];
char labels[33];
```

Προσοχή: Τα στοιχεία των παραπάνω πινάκων είναι a[0] έως a[9], b[0] έως b[29], κλπ.

Πρακτικές συμβουλές για πίνακες

1. Χρειάζεται πραγματικά πίνακας για την επίλυση του προβλήματος; Αν δεν είναι απαραίτητος, δεν τον χρησιμοποιούμε (οι πίνακες ξοδεύουν μεγάλα ποσά μνήμης).
2. Αποφεύγουμε τα πλέον συνήθη λάθη στη χρήση των πινάκων, προσέχοντας πάντα:
   - Να δίνονται αρχικές τιμές σε όλους τους πίνακες.
   - Μην ξεπερνιούνται τα όρια του κάθε πίνακα. Το πιο συνηθισμένο λάθος στη χρήση των πινάκων είναι η προσπάθεια ανάγνωσης ή εκχώρησης τιμής έξω από τα όρια του πίνακα.
   - Η επεξεργασία γίνεται στα στοιχεία του πίνακα (ένα προς ένα). Συνεπώς σε όλες τις εντολές πρέπει να εμφανίζονται τα στοιχεία του πίνακα και όχι το όνομα του ίδιου του πίνακα.
3. Όλα τα στοιχεία του πίνακα έχουν τον ίδιο τύπο, π.χ. όλα είναι ακέραια ή όλα είναι χαρακτήρες όπως ορίστηκαν στο οικείο τμήμα δηλώσεων.
4. Επιλέγουμε και χρησιμοποιούμε την πλέον κατάλληλη μέθοδο για την ταξινόμηση ή την αναζήτηση σε έναν πίνακα.

Υπενθύμιση!

Εισαγωγή δεδομένων σε πίνακα

Παράδειγμα W7.1

```
int i;
float a[30];
for (i=0; i<30; i++)
{
   cout<<“\nAssign value to the ”<<i+1
                              <<“element of the array:  ”;
   cin>>a[i];
}
```

Εκτύπωση των στοιχείων μονοδιάστατου πίνακα

Παράδειγμα

```
#include<iostream>
using namespace std;
int main()
{
    int i;
    float a[30];
```

```
    for (i=0; i<30; i++)
    {
        cout<<"\nAssign value to the "<<i+1<<
                "element of the array:  ";
        cin>>a[i];
    }
    for (i=0; i<30; i++)
        cout<<a[i]<<"   ";
    return 0;
}
```

Παράδειγμα W7.2

Να γραφεί πρόγραμμα το οποίο, θα διαβάζει *100* αριθμούς

$$x_1, x_2, \ldots, x_{100}$$

και στη συνέχεια θα υπολογίζει τους τύπους

$$\mu = \frac{\sum_{i=1}^{100} x_i}{100}$$

και

$$S = \sqrt{\frac{\sum_{i=1}^{100} (x_i - \mu)^2}{99}}$$

```
#include<iostream>
#include<math.h>
using namespace std;
int main()
{
    float x[100],m,s,sum;
    int i;
    cout<<"\nAssign values to the array: ";
```

```
for (i=0; i<100; i++)
{
      cout<<“\nAssign value to the  ”<<i+1
                              <<“ element of the array: ”;
      cin>>x[i];
}
m=0;
for (i=0; i<100; i++)
      m= m+x[i];
m= m/100;
```

```
    sum=0;
    for (i=0; i<100; i++)
        sum=sum+pow((x[i]-m),2);
    s=sqrt(sum/99);
    cout<<"\nm= "<<m;
    cout<<"\ns="<<s;
    return 0;
}
```

Παράδειγμα W7.3

Να γραφεί πρόγραμμα που θα τοποθετεί 100 πραγματικούς αριθμούς σε ένα πίνακα *α* 100 θέσεων και στη συνέχεια θα υπολογίζει το

$$\sum_{i=0}^{99} \frac{\eta\mu^2(a(i))}{a(i)^3+1}$$

ΠαράδειγμαW7.4

Να γράψετε πρόγραμμα που θα διαβάζει δύο μονοδιάστατους πίνακες και θα βρίσκει το άθροισμά τους. Το πρόγραμμα θα τυπώνει τους δυο πίνακες και τον πίνακα που προκύπτει από το άθροισμά τους.

Παράδειγμα W7.5

Να γράψετε πρόγραμμα που θα διαβάζει 10 ακέραιους αριθμούς και θα τους τοποθετεί σε ένα πίνακα 10 θέσεων. Στη συνέχεια θα υπολογίζει το πλήθος των αριθμών που είναι πολλαπλάσια του 3 καθώς και το γινόμενο των θετικών ακεραίων των οποίων το παραγοντικό είναι μεγαλύτερο του 2.01E103.

```
#include<iostream>
#include<math.h>
using namespace std;
int main()
{
    int x[10],i, j, a, plithos, product;
    double fact;
    cout<<“\nAssign values to the array:”;
```

```
for (i=0; i<10; i++)
{
      cout<<“\nAssign value to the  ”<<i+1
                        <<“ element of the array: ”;
      cin>>x[i];
}
plithos=0;
j=0;
product =1;
for (i=0; i<10; i++)
{
      if (x[i]%3==0)
      plithos=plithos+1;
```

```
        if (x[i]>0)
        {
                fact=1;
                for (a=1; a<=x[i]; a++)
                        fact=fact*a;
                if (fact>2.01E103)
                {
                        j=1;
                        product=product*x[i];
                }
         }
}
```

```
    cout<<“The number of multiples of 3 is: ”<<plithos;
    if (j==0)
        cout<<“\nThere are no integers with factorial
                                greater than 2.01E103”;
    else
        cout<<“The product of integers with factorial
                greater than 2.01E103 is:  ”<<product;
    return 0;
}
```

Πίνακας με δυναμικό μέγεθος

Χρησιμοποιώ μεταβλητή τύπου «δείκτη». Ο δείκτης:

- Δείχνει προς τα στοιχεία του δυναμικού πίνακα.
- Περιλαμβάνει τον τύπο δεδομένων των στοιχείων αυτών στη δήλωση του.

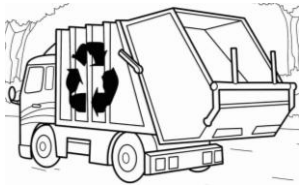
Οι τελεστές (για δείκτες)

new για να δεσμεύσω μνήμη (η οποία διατίθεται για τα στοιχεία του πίνακα)



https://en.cppreference.com/cpp/language/new

delete για να αποδεσμεύσω τη μνήμη αυτή (όταν πλέον δεν την χρειάζομαι)



https://en.cppreference.com/cpp/language/delete

είναι <u>απαραίτητοι</u>!

Πίνακας με δυναμικό μέγεθος

```
#include<iostream>
#include<math.h>
using namespace std;
int main()
{
int* a = NULL;   // Pointer to int, initialize to nothing.
int i;          // index for the array
int n;          // Size needed for array
cout << "Please specify size of the array (give a positive integer):";
cin >> n;       // Read in the size
a = new int[n];  // Allocate n ints and save ptr in a.
for (int i=0; i<n; i++) {
    a[i] = 0;    // Initialize all elements to zero.
}
//. . .  // Use a as a normal array
for (i=0; i<n; i++)
                a[i] = pow(i,2);
for (i=0; i<n; i++)
                cout<<"\n"<<"a["<<i<<"]="<<a[i];
delete [] a;  // When done, free memory pointed to by a.
a = NULL;     // Clear a to prevent using invalid memory reference.
return 0;
}
```