

## ΠΙΝΑΚΕΣ ΣΤΗΝ ΓΛΩΣΣΑ C

Ένας **πίνακας** είναι ένα σύνολο μεταβλητών του ίδιου τύπου. Το κάθε στοιχείο του πίνακα αναγνωρίζεται από ένα ακέραιο δείκτη (index). Στη C ο δείκτης θέσης αρχίζει από το μηδέν (0) και γράφεται πάντα μέσα σε τετράγωνα αγκύλες ( [ και ] )

Έχουμε ήδη συναντήσει μονοδιάστατους πίνακες, οι οποίοι ορίζονται ως εξής:

```
int results[20];
```

Οι πίνακες μπορούν να έχουν περισσότερες από μια διαστάσεις, οπότε ορίζονται ως:

```
int results_2d[20][5];
```

```
int results_3d[20][5][3];
```

Κάθε δείκτης έχει τις δικές του τετράγωνα αγκύλες.

Όταν ένας πίνακας ορίζεται στη main συνήθως συμπεριλαμβάνει πληροφορίες για τις διαστάσεις του. Είναι δυνατόν να χρησιμοποιηθεί ένας άλλος τύπος που λέγεται pointer (δείκτης - όχι index). Αυτό σημαίνει ότι οι διαστάσεις δεν καθορίζονται αμέσως, αλλά μπορεί να δεσμευτεί χώρος όταν χρειαστεί. Είναι μια προχωρημένη τεχνική που χρησιμοποιείται μόνο σε ορισμένα εξειδικευμένα προγράμματα.

Όταν περνά σε μία συνάρτηση ως παράμετρος, η συνάρτηση αυτή δεν χρειάζεται να γνωρίζει το μέγεθος του πίνακα. Για παράδειγμα αν έχουμε μία συνάρτηση που ταξινομεί μια λίστα (η οποία αναπαριστάται από ένα πίνακα) τότε η συνάρτηση μπορεί να ταξινομεί λίστες διαφόρων μεγεθών. Το μειονέκτημα είναι ότι η συνάρτηση δεν μπορεί να αντιληφθεί το μέγεθος της λίστας, οπότε αυτή η πληροφορία πρέπει να περαστεί ως πρόσθετη παράμετρος.

## ΠΑΡΑΔΕΙΓΜΑΤΑ ΜΕ ΠΙΝΑΚΕΣ

### ΠΑΡ 1

#### ΠΡΟΓΡΑΜΜΑ ΔΙΑΒΑΣΜΑΤΟΣ-ΕΚΤΥΠΩΣΗΣ ΚΑΙ ΕΥΡΕΣΗΣ ΤΟΥ ΜΕΣΟΥ ΟΡΟΥ ΣΤΟΙΧΕΙΩΝ ΜΟΝΟΔΙΑΣΤΑΤΟΥ ΠΙΝΑΚΑ Ν ΑΚΕΡΑΙΩΝ

Σχόλιο: Οποιαδήποτε λειτουργία σε μονοδιάστατο πίνακα απαιτεί τη χρήση της εντολής for

```
#include <stdio.h>
```

```
#define N 5; //Δηλώνουμε τη σταθερά N με τιμή 5
```

```
void main()  
{
```

```
    int sum=0,i,x[N]; //Το μέγεθος ενός πίνακα είναι πάντα σταθερό.  
    float av;
```

//Το i δείχνει πάντα τη θέση του πίνακα ενώ το x[i] δείχνει το στοιχείο του πίνακα στη θέση i. Τα στοιχεία ενός πίνακα αριθμούνται πάντα από το 0

//και συμβολίζονται x[0], x[1], x[2], x[3] και x[4].

```

for (i=0;i<=N-1;i++)//To for αυτό υλοποιεί το διάβασμα του πίνακα
{
    printf("Dose to %d stoixeio: ",i+1);
    scanf("%d",&x[i]);
}

printf("\nPinakas\n");

for (i=0;i<=N-1;i++)//To for αυτό υλοποιεί την εκτύπωση του πίνακα
    printf("%d\t",x[i]); //Ο χαρακτήρας \t συμβολίζει το tab
printf("\n");//Αλλαγή γραμμής

for (i=0;i<=N-1;i++)// εύρεση αθροίσματος των στοιχείων του πίνακα
    sum+=x[i]; //το ίδιο ακριβώς με το sum=sum+x[i];

av=(float)sum/N; //Εύρεση μέσου όρου του πίνακα.

// Το sum μετατρέπεται προσωρινά σε float για να υπολογίσουμε δεκαδικό μέσο όρο
//Ο μέσος όρος υπολογίζεται μόνο 1 φορά και γιαυτό τοποθετείται εκτός της επανάληψης.

printf("\nO mesos oros tou pinaka = %.2f\n",av);
}

```

**ΠΑΡ2**  
**ΠΡΟΓΡΑΜΜΑ ΔΙΑΒΑΣΜΑΤΟΣ-ΕΚΤΥΠΩΣΗΣ ΚΑΙ ΤΑΞΙΝΟΜΗΣΗΣ ΤΩΝ ΣΤΟΙΧΕΙΩΝ**  
**ΜΟΝΟΔΙΑΣΤΑΤΟΥ ΠΙΝΑΚΑ Ν ΑΚΕΡΑΙΩΝ ΣΕ ΑΥΞΟΥΣΑ ΣΕΙΡΑ**  
**ΣΥΜΦΩΝΑ ΜΕ ΤΗ ΜΕΘΟΔΟ ΤΗΣ ΦΥΣΑΛΙΔΑΣ**

```

#include <stdio.h>

#define N 5 ; //Δηλώνουμε τη σταθερά N με τιμή 5

void main()
{

    int k, temp, i, x[N];//Το μέγεθος ενός πίνακα είναι πάντα σταθερό.

    for (i=0;i<=N-1;i++)//To for αυτό υλοποιεί το διάβασμα του πίνακα
    {
        printf("Dose to %d stoixeio: ",i+1);
        scanf("%d",&x[i]);
    }

    printf("\nPinakas\n");
    for (i=0;i<=N-1;i++)//To for αυτό υλοποιεί την εκτύπωση του πίνακα
        printf("%d\t",x[i]); //Ο χαρακτήρας \t συμβολίζει το tab
    printf("\n");//Αλλαγή γραμμής
}

```

```

//Τα 2 for υλοποιούν την ταξινόμηση φυσαλίδας
for (k=1;k<=N-1;k++) //Το k είναι ο μετρητής των περασμάτων του πίνακα.

    for (i=0;i<N-k;i++) //Το for δίνει τις συγκρίσεις N-k που γίνονται σε κάθε πέρασμα.
        if (x[i]>x[i+1]) //Στο if αυτό γίνεται η εναλλαγή των στοιχείων.

            {
                temp=x[i];
                x[i]=x[i+1];
                x[i+1]=temp;
            }

printf("\nTaxinomimenos Pinakas\n");

for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί την εκτύπωση του πίνακα
    printf("%d\t",x[i]); //Ο χαρακτήρας \t συμβολίζει το tab
printf("\n");//Αλλαγή γραμμής
}

```

### ΠΑΡ3

#### ΠΡΟΓΡΑΜΜΑ ΕΥΡΕΣΗΣ ΤΟΥ ΜΕΓΑΛΥΤΕΡΟΥ ΚΑΙ ΤΟΥ ΜΙΚΡΟΤΕΡΟΥ ΣΤΟΙΧΕΙΟΥ ΕΝΟΣ ΜΟΝΟΔΙΑΣΤΑΤΟΥ ΠΙΝΑΚΑ Ν ΑΚΕΡΑΙΩΝ ΚΑΘΩΣ ΚΑΙ ΤΩΝ ΘΕΣΕΩΝ ΤΟΥΣ ΣΤΟΝ ΠΙΝΑΚΑ

```

#include <stdio.h>

#define N 5 ; //Δηλώνουμε τη σταθερά N με τιμή 5

void main()
{

    int max, maxp, min, minp ,i ,x[N];

    for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί το διάβασμα του πίνακα

        {
            printf("Dose to %d stoixeio: ",i+1);
            scanf("%d",&x[i]);
        }

    printf("\nPinakas\n");
    for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί την εκτύπωση του πίνακα
        printf("%d\t",x[i]); //Ο χαρακτήρας \t συμβολίζει το tab
    printf("\n");//Αλλαγή γραμμής

    max=min=x[0];//Υποθέτω ότι το μέγιστο και το ελάχιστο στοιχείο του πίνακα είναι το 1ο
    maxp=minp=0;//Υποθέω ότι η θέση του max και του min είναι η πρώτη δηλαδή το 0

    for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί την εύρεση του μέγιστου και του ελάχιστου

```

```

{
    if (x[i]>max) //Το if αυτό βρίσκει το μέγιστο στοιχείο και τη θέση του στον πίνακα
    {
        max=x[i];
        maxp=i;
    }

    if (x[i]<min) //Το if αυτό βρίσκει το ελάχιστο στοιχείο και τη θέση του στον πίνακα
    {
        min=x[i];
        minp=i;
    }
}
printf("Megisto stoixeio pinaka = %d sti thesi %d \n",max,maxp);
printf("Elaxisto stoixeio pinaka = %d sti thesi %d \n",min,minp);
}

```

#### ΠΑΡ4

#### ΠΡΟΓΡΑΜΜΑ ΣΕΙΡΙΑΚΗΣ ΑΝΑΖΗΤΗΣΗΣ ΣΤΟΙΧΕΙΟΥ ΣΕ ΜΟΝΟΔΙΑΣΤΑΤΟ ΠΙΝΑΚΑ Ν ΑΚΕΡΑΙΩΝ ΚΑΙ ΕΚΤΥΠΩΣΗΣ ΟΛΩΝ ΤΩΝ ΘΕΣΕΩΝ ΤΟΥ ΣΤΟΝ ΠΙΝΑΚΑ ΚΑΘΩΣ ΚΑΙ ΤΟΥ ΠΛΗΘΟΥΣ ΤΩΝ ΕΜΦΑΝΙΣΕΩΝ ΤΟΥ ΣΤΟΝ ΠΙΝΑΚΑ

```
#include <stdio.h>
```

```
#define N 5 ; //Δηλώνουμε τη σταθερά N με τιμή 5
```

```
void main()
{
```

```
    int m=0, s, i, x[N];
```

```
    for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί το διάβασμα του πίνακα
    {
        printf("Dose to %d stoixeio: ",i+1);
        scanf("%d",&x[i]);
    }

```

```
    printf("\nPinakas\n");
    for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί την εκτύπωση του πίνακα
        printf("%d\t",x[i]); //Ο χαρακτήρας \t συμβολίζει το tab
    printf("\n");//Αλλαγή γραμμής

```

```
printf("Dose to stoixeio anazitisis\n");
scanf("%d",&s); //Διαβάζουμε το στοιχείο αναζήτησης στη μεταβλητή s

```

```
for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί τη σειριακή αναζήτηση στον πίνακα
```

```
    if (x[i]==s)
```

```

        {
            printf("To stoixeiio brethike sti thesi %d\n",i);//Τυπώνεται η κάθε θέση που //το
                                                    στοιχείο βρέθηκε στον πίνακα
            m++;//Αυξάνεται ο μετρητής των εμφανίσεων του στοιχείου στον πίνακα
        }

if (m==0)
    printf("To stoixeiio %d den brethike ston pinaka\n",s);
else
    printf("To stoixeiio %d brethike synolika %d fores ston pinaka\n",s,m);
}

```

## ΠΑΡ5 ΠΡΟΓΡΑΜΜΑ ΔΥΑΔΙΚΗΣ ΑΝΑΖΗΤΗΣΗΣ ΣΤΟΙΧΕΙΟΥ ΣΕ ΜΟΝΟΔΙΑΣΤΑΤΟ ΠΙΝΑΚΑ Ν ΑΚΕΡΑΙΩΝ

```

#include <stdio.h>

#define N 5; //Δηλώνουμε τη σταθερά N με τιμή 5

void main()
{
    int k,temp,left,right,meso,found,s,i,x[N];

    for (i=0;i<N;i++)//Το for αυτό υλοποιεί το διάβασμα του πίνακα
    {
        printf("Dose to %d stoixeiio: ",i+1);
        scanf("%d",&x[i]);
    }

    printf("\nPinakas\n");
    for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί την εκτύπωση του πίνακα
        printf("%d\t",x[i]); //Ο χαρακτήρας \t συμβολίζει το tab δ        printf("\n");//Αλλαγή
γραμμής

    //Τα 2 for υλοποιούν την ταξινόμηση φουσαλίδας
    for (k=1;k<=N-1;k++) //Το k είναι ο μετρητής των περασμάτων του πίνακα.

        for (i=0;i<N-k;i++) //Το for αυτό δίνει τις συγκρίσεις που γίνονται σε κάθε πέρασμα.
            if (x[i]>x[i+1]) //Στο if αυτό γίνεται η εναλλαγή των στοιχείων.
            {
                temp=x[i];
                x[i]=x[i+1];
                x[i+1]=temp;
            }

    printf("\nTaxinomimemos Pinakas\n");
    for (i=0;i<=N-1;i++)//Το for αυτό υλοποιεί την εκτύπωση του πίνακα
        printf("%d\t",x[i]); //Ο χαρακτήρας \t συμβολίζει το tab

```

```

printf("\nDose to stoixeiio anazitisis\n");
scanf("%d",&s); //Διαβάζουμε το στοιχείο αναζήτησης στη μεταβλητή s

left=0; //Η μεταβλητή left δείχνει το κάτω όριο του πίνακα
right=N-1; //Η μεταβλητή right δείχνει το επάνω όριο του πίνακα
found=0; //Υποθέτουμε αρχικά ότι το αναζητούμενο στοιχείο δεν υπάρχει στον πίνακα

while (left<=right && found==0) // όσο δεν έχουμε ελέγξει όλα τα στοιχεία και το s δεν έχει ακόμα
                                                                    βρεθεί στον πίνακα επανέλαβε
{
    meso=(left+right)/2; //Στο meso καταχωρείται η θέση του μεσαίου στοιχείου

    if (x[meso]==s) //Αν το αναζητούμενο s είναι στο μεσαίο στοιχείο του πίνακα
    {
        printf("To stoixeiio %d brethike sth thesi %d tou pinaka\n",s,meso);
        found=1; //Θέτουμε στη found την τιμή 1 για να σταματήσει η επανάληψη
    }

    else
        if (x[meso]<s) //Αν το αναζητούμενο s είναι μεγαλύτερο από το μεσαίο
                                                                    στοιχείο του πίνακα τότε η αναζήτηση
                                                                    //περιορίζεται στο επάνω μισό του πίνακα

            left=meso+1;
            else //Αν το αναζητούμενο s είναι μικρότερο από το μεσαίο στοιχείο του
                                                                    πίνακα τότε η αναζήτηση
                                                                    //περιορίζεται στο κάτω μισό του πίνακα

                right=meso-1;
}

if (found==0)
    printf("To stoixeiio %d den brethike ston pinaka\n",s);
}

```