

Αλφαριθμητικά

Οι πίνακες χαρακτήρων που έχουν ως τελευταίο στοιχείο τους το χαρακτήρα '\0' ονομάζονται αλφαριθμητικά (strings).

Τα αλφαριθμητικά είναι και αυτά πίνακες χαρακτήρων , αλλά έχουν και επιπλέον χαρακτηριστικά . Με την προσθήκη του χαρακτήρα τέλους '\0' μπορώ να χρησιμοποιήσω ένα πλήθος έτοιμων συναρτήσεων κάνοντας προσθήκη ενός αρχείου επικεφαλίδας #include<string.h>

char ship[4]={‘Z’,’E’,’Y’,’S’};

Εδώ έχω έναν πίνακα χαρακτήρων με όνομα ship και πλήθος στοιχείων 4 , δηλαδή από θέση 0 έως και θέση 3. Δεν είναι συμβολοσειρά-αλφαριθμητικό

Ενώ αν γράψω :

Char ship[5]={‘Z’,’E’,’Y’,’S’,’\0’}; τότε θεωρείται αλφαριθμητικό . Το πρόγραμμα πρέπει να βλέπει το χαρακτήρα τέλους '\0' για να καταλαβαίνει ότι πρόκειται για αλφαριθμητικό οπότε και μπορεί να το βλέπει σαν να είναι μία λέξη και όχι μόνο σαν ένας απλός πίνακας χαρακτήρων.

Τα αλφαριθμητικά δηλαδή έχουν τα χαρακτηριστικά των πινάκων χαρακτήρων με κάποια άλλα επιπλέον όμως.

Πώς αρχικοποιώ ένα αλφαριθμητικό;

Char ship[5]={‘Z’,’E’,’Y’,’S’,’\0’};

Char ship[]="ZEYS";

Με την παραπάνω εντολή μπαίνει αυτόματα ο χαρακτήρας τέλους και ρυθμίζεται αυτόματα και το μήκος του πίνακα

Το ίδιο κάνει και η εντολή:

Char ship[5]="ZEYS";

Αν γράψω

Char ship[8]="ZEYS"; τότε οι θέσεις 0..και 3 έχουν τους χαρακτήρες της λέξης ZEYS ενώ στις θέσεις 4,5,6,7 υπάρχει αυτόματα ο τερματικός χαρακτήρας '\0'.

Πώς δίνω τιμή σε ένα αλφαριθμητικό από το πληκτρολόγιο και πως το εμφανίζω.

```
name_string.c
1 #include<stdio.h>
2 void main()
3 {
4     char name[8];
5     printf("Pliktrologiste to mikro sas onoma\n");
6     scanf("%7s",name);
7     printf("%s",name);
8     printf("\nPliktrologises th lexi:%s",name);
9
10}
11
```

Συντόμευση για το jfe.exe
Pliktrologiste to mikro sas onoma
mihalis
mihalis
Pliktrologises th lexi:mihalis

Στο παραπάνω πρόγραμμα δηλώνω μήκος name=8 . Δηλαδή θέλω 7 στοιχεία για το όνομα μου και ένα για τον τερματικό χαρακτήρα. Θα μπορούσα να μη βάλω την εντολή scanf("%7s",name) που διαβάζει το πολύ 7 χαρακτήρες από το πληκτρολόγιο , αλλά την εντολή scanf("%s",name) αλλά αν πληκτρολογούσα μεγαλύτερο όνομα δεν θα υπήρχε χώρος για τον τερματικό χαρακτήρα. Παρατηρώ ότι στη scanf() δεν βάζω το χαρακτήρα & γιατί πρόκειται για εκχώρηση σε αλφαριθμητικό και το όνομα name είναι από μόνο του διεύθυνση.

Εναλλακτικά το πιο πάνω με τις εντολές get(s) και put(s). Η gets() διαβάζει χαρακτήρες από το πληκτρολόγιο και τους αποθηκεύει στον πίνακα trash προσθέτοντας και το χαρακτήρα τέλους. Η puts() εμφανίζει στην οθόνη το αλφαριθμητικό. Αποφεύγονται να χρησιμοποιούνται γιατί μπορεί να μας οδηγήσουν σε σφάλματα αν το μέγεθος του πίνακα δεν επαρκεί . (Γι' αυτό και επιλέγεται η fgets() που μπορούμε να ορίσουμε μέγεθος)

```
(All global members) main
#include<stdio.h>
void main()
{
    char trash[18];
    int i=0,size=0;
    printf("Pliktrologiste mia frasi\n");
    gets(trash);
    puts(trash);
    printf("\nPliktrologises th trash:%s",trash);
    while(trash[i]!='\0')
    {
        size++;
        i++;
    }
    printf("\nSize=%d\n",size);
}
```

C:\Program Files\Microsoft Visual Studio\MyP... Pliktrologiste mia frasi
Exei kaih mera
Exei kaih mera
Pliktrologises th trash:Exei kaih mera
Size=14
Press any key to continue

Πώς μετράω μήκος αλφαριθμητικού (συμβολοσειράς.)

Παίρνω τον πίνακα που έχω και τον διατρέχω μέχρι να βρω τον τερματικό χαρακτήρα '\0' μετρώντας το πλήθος των στοιχείων του

The screenshot shows the Microsoft Visual Studio IDE. In the top window, titled 'main', the code is:

```
#include<stdio.h>
void main()
{
    char name[8];
    int i=0, size=0;
    printf("Pliktrologiste to mikro sas onoma\n");
    scanf("%7s", name);
    printf("%s", name);
    printf("\nPliktrologises th lexi:%s", name);
    while(name[i]!='\0')
    {
        size++;
        i++;
    }
    printf("\nSize=%d\n", size);
}
```

In the bottom window, titled 'C:\Program Files\Microsoft Visual Studio\...', the output is:

```
Pliktrologiste to mikro sas onoma
Nikos
Nikos
Pliktrologises th lexi:Nikos
Size=5
Press any key to continue...
```

Βασικές συναρτήσεις αλφαριθμητικών

- **strcpy(s1,s2)**

Η εντολή αυτή αντιγράφει το αλφαριθμητικό s2 (ή αλφαριθμητική σταθερά) σε ένα πίνακα χαρακτήρων s1 . Αντιγράφεται και ο χαρακτήρας τέλους από το αλφαριθμητικό s2 και μας επιστρέφει το s1

Παράδειγμα

ΝΓΠ που να δέχεται το μικρό μας όνομα από το πληκτρολόγιο και να το εκγρωρεί σε μια συμβολοσειρά με όνομα name. Στη συνέχεια να το αντιγράφει σε μία συμβολοσειρά με όνομα FirstName και να εμφανίζει το FirstName

```
#include<stdio.h>
#include<string.h>
void main()
{
    char name[10], FirstName[10];
    printf("Pliktrologise to mikro onoma sou\n");
    scanf("%9s", name);
    strcpy(FirstName, name);
    printf("\nTo FirstName einai::%s", FirstName);
    printf("\nTo FirstName einai::%s\n", strcpy(FirstName, name));
}

C:\Program Files\Microsoft Visual Studio\... □ X
Pliktrologise to mikro onoma sou
Nikos
To FirstName einai::Nikos
To FirstName einai::Nikos
Press any key to continue
```

- **strlen()**

Η εντολή αυτή μας επιστρέφει το μήκος σε bytes της συμβολοσειράς
(>αλφαριθμητικό ή string)

Παράδειγμα:

**ΝΓΠ που να δέχεται ένα όνομα από το πληκτρολόγιο μεγίστου πλήθους
χαρακτήρων 9 και να εμφανίζει στην οθόνη το όνομα καθώς και το μήκος
του ονόματος**

```
#include<stdio.h>
#include<string.h>
void main()
{
    char name[10];
    printf("Pliktrologise to mikro onoma sou\n");
    scanf("%9s", name);
    printf("\nTo name einai::%s", name);
    printf("\nTo megethos tou einai:%d\n", strlen(name));
    printf("\nTo megethos tou pinaka einai:%d\n", sizeof(name));
}

C:\Program Files\Microsoft Visual Studio\MyProje... □ X
Pliktrologise to mikro onoma sou
Xara
To name einai::Xara
To megethos tou einai:4
To megethos tou pinaka einai:10
Press any key to continue
```

- **strcat(s1,s2)**

Η συνάρτηση αυτή προσθέτει στο τέλος του s1 το αλφαριθμητικό s2. Το πρώτο στοιχείο του προστιθέμενου αλφαριθμητικού s2 γράφεται πάνω στο χαρακτήρα τέλους του s1. Η συνάρτηση μας επιστρέφει το νέο s1

Παράδειγμα

ΝΓΠ που να δέχεται δύο συμβολοσειρές . Η s1 θα έχει τιμή «New» και η s2 θα έχει τιμή «York». Με χρήση της κατάλληλης συνάρτησης strcat(s1,s2) να τις συνενώσετε και να εμφανίσετε το νέο συνενωμένο περιεγόμενο της s1. (Προσέχω στη συνένωση γιατί το πρώτο στοιχείο του προστιθέμενου αλφαριθμητικού s2 γράφεται πάνω στο χαρακτήρα τέλους του s1)

The screenshot shows the Microsoft Visual Studio IDE. In the top pane, there is a code editor with the following C code:

```
#include<stdio.h>
#include<string.h>
void main()
{
    char s1[]="New ";
    char s2[]="York";
    printf("s1=%s\n",strcat(s1,s2));
}
```

In the bottom pane, there is a terminal window titled "C:\Program Files\Microsoft Vis...". It displays the output of the program: "s1=New York". Below the output, there is a message: "Press any key to continue...".

- **strcmp(s1,s2)**

Η συνάρτηση αυτή συγκρίνει τις δύο συμβολοσειρές s1 και s2.

Αν το s1<s2 επιστρέφει αρνητικό. Αν είναι s1==s2 επιστρέφει 0 και αν s1>s2 επιστρέφει θετικό. Όταν μιλάμε για σύγκριση εννοούμε λεξικογραφική σύγκριση σύμφωνα με τον πίνακα ASCII

Παράδειγμα .

Να γίνει πρόγραμμα που να δέχεται δύο επώνυμα από το πληκτρολόγιο και να εμφανίζει σχετικό μήνυμα αν είναι ίδια , διαφορετικά να μας τα εμφανίζει σε αλφαριθμητική σειρά.

```
#include<stdio.h>
#include<string.h>
void main()
{
char s1[10],s2[10];
printf("S1=?\n");
scanf("%9s",s1);
printf("S2=?\n");
scanf("%9s",s2);
((strcmp(s1,s2)==0)?printf("Idia\n"):(strcmp(s1,s2)<0)?printf("\n%s %s",s1,s2):printf("\n%s %s",s2,s1)));
if(strcmp(s1,s2)==0)
{
    printf("Idia\n");
}
else if(strcmp(s1,s2)<0)
{
    printf("\n%s %s",s1,s2);
}
else
{
    printf("\n%s %s",s2,s1);
}
}
```

"C:\Program Files\Microsoft Visual Studio\MyProjects\strings\Debug" □ x

```
S1=?
Xara
S2=?
Katerina
Katerina Xara
Katerina XaraPress any key to continue...
```

Γράψτε ένα πρόγραμμα στο οποίο να ορίσετε μία συμβολοσειρά (s1) και να αποθηκεύσετε σε αυτή την φάση **ATEI**.

Τυπώστε την συμβολοσειρά (s1) και τον αριθμό των χαρακτήρων που περιέχει.

Κατασκευάστε μια νέα συμβολοσειρά (s2) που να περιέχει την φάση **IONIΩΝ ΝΗΣΩΝ**.

Να γίνει η συνένωση της s1 με την s2 σε μια νέα συμβολοσειρά s3.

Τυπώστε την νέα συμβολοσειρά s3 και τον αριθμό των χαρακτήρων που περιέχει.

```

1 #include<stdio.h>
2 #include<string.h>
3 void main()
4 {
5 char s1[5]={"ATEI"};
6 char s2[14]=" IONION NHSON";//char s2[]="IONION NHSON";
7 char s3[20];
8 printf("%s\n",s1);
9
10 /*ENALLAKTIKA*/
11 puts(s1);
12
13 printf("\nPlithos stoixeion symboloseiras s1=%d\n",strlen(s1));
14 printf("Synenosi s1:%s+s2:%s = \n",s1,s2);
15 strcpy(s3,strcat(s1,s2));
16 printf("%s",s3);
17 printf("\nSize=%d",strlen(s3));
18 }
19

```

jfe - Συντόμευση

```

ATEI
ATEI

Plithos stoixeion symboloseiras s1=4
Synenosi s1:ATEI+s2: IONION NHSON =
ATEI IONION NHSON
Size=17_

```

Συναρτήσεις συμβολοσειρών: strcat

```

#include <stdio.h>
#include <string.h>
main()
{
    char str1[]={'H','e','l','l','o','\0'};
    char str2[]=" John";
    printf ("%os + %os = %os\n", str1, str2, strcat(str1,str2));
}

```

Hello John + John = Hello John

Το λάθος στο παραπάνω πρόγραμμα είναι ότι παίρνει ένα αλφαριθμητικό str1 και το συνενώνει με το str2 και εμφανίζει το νέο str1 σύμφωνα με όσα μάθαμε για τη συνάρτηση strcat(str1,str2). **Στην οθόνη ΔΕΝ εμφανίζεται το Hello + John=Hello John όπως θα περιμέναμε , αλλά το Hello John + John =Hello John. Γιατί;**

A screenshot of a C programming environment. The code editor window contains the following C code:

```
1 #include <stdio.h>
2 #include <string.h>
3 main(){
4     char str1[]={'H','e','l','l','o','\0'};
5     char str2[]=" John";
6     printf ("%s + %s = %s\n", str1, str2, strcat(str1,str2));
7 }
8
```

The terminal window below shows the output:

```
jfe - Συντόμευση
Hello John + John = Hello John
```

Αλλάζοντας λίγο τον κώδικα

A screenshot of a C programming environment showing the same code as the first one, but with a modification in the printf statement:

```
1 #include <stdio.h>
2 #include <string.h>
3 main(){
4     char str1[]={'H','e','l','l','o','\0'};
5     char str2[]=" John";
6     printf ("%s + %s = ", str1, str2);
7     printf("%s\n",strcat(str1,str2));
8 }
```

The terminal window shows the output:

```
jfe - Συντόμευση
Hello + John = Hello John
```

Επειδή καλείται στη γραμμή 6 πρώτα η εντολή strcat(str1,str2) αλλάζει τη τιμή της str1 οπότε εμφανίζεται αλλοιωμένη στην εντολή printf()