

Η γλώσσα προγραμματισμού C



**Χειρισμός χαρακτήρων,
συμβολοσειρές**

Κ. Βασιλάκης, ΣΤΕΦ, ΤΕΙ Κρήτης

Συμβολοσειρές

- Συμβολοσειρά ονομάζουμε μια οποιαδήποτε ακολουθία αλφαριθμητικών χαρακτήρων:

«Κώστας», «Κώστας Βασιλάκης», «Δαιδάλου 23»

- Στην C δεν υπάρχει ιδιαίτερος τύπος για τον χειρισμό των συμβολοσειρών.
- Για την αποθήκευση τους χρησιμοποιούμε την δομή του πίνακα τύπου `char` και φυσικά τις διαχειριζόμαστε όπως ένα πίνακα:

`char line[80];` // δήλωση συμβολοσειράς με 80 χαρακτήρες

`line[i] = getchar();` // απόδοση τιμής σε στοιχείο της σειράς

- Αν πρέπει λοιπόν να διαχειριστούμε N χαρακτήρες ο πίνακας της συμβολοσειράς θα πρέπει να δηλωθεί με N+1 χαρακτήρες.
- Υπάρχουν ειδικές συναρτήσεις για τη διαχείριση των συμβολοσειρών (`#include <string.h>`)



Παρατηρήσεις για την αποθήκευση συμβολοσειρών

- Ο τελευταίος χαρακτήρας μιας συμβολοσειράς είναι ο «\0» (μηδενικός χαρακτήρας – null character).
- Ο μηδενικός χαρακτήρας «\0» είναι διαφορετικός από τον χαρακτήρα 0 (μηδέν). Ο μηδενικός χαρακτήρας έχει την τιμή 0 (εσωτερική αναπαράσταση:00000000), ενώ ο χαρακτήρας «0» έχει την τιμή 48 (00110000). Δηλαδή η εντολή:

printf(" %d, %d", '\0','0'); // θα δώσει σαν αποτέλεσμα: 0, 48

- Οι τιμές που δίνουμε στις μεταβλητές που έχουν δηλωθεί σαν συμβολοσειρές, οριοθετούνται με λατινικά εισαγωγικά διπλά (") ή απλά ('). Προσοχή το 'A' δεν είναι ίσο με το "A" :

Το 'A' χρειάζεται 1 byte μνήμης (01000001)

Το "A" χρειάζεται 2 bytes ('A','\0' ==> 0100000100000000)

- Το "A" είναι μία συμβολοσειρά που περιέχει το γράμμα 'A' και στο τέλος τον χαρακτήρα '/0' .



Αρχικές τιμές (με τη δήλωση)

- Όπως στους μονοδιάστατους πίνακες:

```
char str1[6] = {'h', 'e', 'l', 'l', 'o', '\0'};
```

(Το `str1[0]` είναι `h`, το `str1[1]` είναι `e` και το `str1[5]` είναι `'\0'`)

- Εναλλακτικά:

```
char str2[8] = "hello"; //μπαίνει αυτόματα το '\0' στο τέλος
```

- Επίσης:

```
char str2[] = "hello"; //το πλήθος υπολογίζεται αυτόματα
```

- Αν δώσουμε: **`char str[20] = "hello";`** δεσμεύουμε 20 θέσεις μνήμης ενώ χρειαζόμαστε μόνο 6. Οι θέσεις που περισσεύουν παίρνουν τη τιμή `'\0'` .
- Η εντολή **`char str[20] = {'\0'};`** δίνει την τιμή `'\0'` σε όλα τα στοιχεία της συμβολοσειράς **`str`**.
- Η **`sizeof(str)`** επιστρέφει το μέγεθος όλου του πίνακα.



Διάβασμα από το πληκτρολόγιο – `scanf()`, `gets()`

- Με την γνωστή `scanf()` με χρήση του προσδιοριστή `%s`:

```
printf("What's your name :"); scanf("%s", str);
```

 - Προσοχή: δεν βάζουμε τον χαρακτήρα `&` μπροστά από το `str` διότι το `str` είναι η διεύθυνση του 1ου στοιχείου της συμβολοσειράς.
 - Επίσης προσοχή: Η `scanf()` σταματά τη ανάγνωση μόλις συναντήσει τον κενό χαρακτήρα (space) ή το χαρακτήρα ελέγχου για τη αλλαγή γραμμής (`\n`).
- Εναλλακτικά, για το διάβασμα μιας συμβολοσειράς από το πληκτρολόγιο μπορούμε να χρησιμοποιήσουμε τη συνάρτηση `gets()`:

```
printf("What's your name:"); gets(name);
```
- Η `gets()` διαβάζει συνεχώς από το πληκτρολόγιο μέχρι να συναντήσει τον χαρακτήρα `\n` (ακόμα και τους κενούς χαρακτήρες). Αντικαθιστά δε τον `\n` με τον `\0`.



Εμφάνιση συμβολοσειρών στη οθόνη

- Με την συνάρτηση ***printf()*** και τον προσδιοριστή %s:
printf("To onoma sou einai: %s\n", onoma);
 - Η printf() εμφανίζει όλους τους χαρακτήρες μέχρι να συναντήσει το μηδενικό χαρακτήρα '\0'.
- Με την printf() μπορούμε να εμφανίσουμε μέρος της συμβολοσειράς:
printf("To onoma sou einai: %s\n", onoma+2);
(εμφανίζει την συμβολοσειρά από τον 3^ο χαρακτήρα και μετά)
- Με τη συνάρτηση ***puts()***:
printf("What's your name:"); gets(name);
printf("Hello "); puts(name);
 - Η puts() μετά την εμφάνιση της συμβολοσειράς, τυπώνει τον χαρακτήρα '\n'.



Τρεις διαφορετικοί τρόποι να διαβάσουμε μια συμβολοσειρά

```
#include <stdio.h>
main() { // με scanf() & printf()
    char line [20];
    printf("Enter line: ");
    scanf("%s", line);
    printf("Line: %s\n", line);
}
```

```
#include <stdio.h>
main() { // με gets() & puts()
    char line[30];
    printf("Enter line: ");
    gets(line);
    printf("Line: ");
    puts(line);
}
```

```
#include <stdio.h>
main() { // με την getchar();
    char line[30], ch;
    int i = 0;
    printf("Enter line: ");
    while(ch != '\n') { //μέχρι το newline
        ch = getchar();
        line[i] = ch;
        i++;
    }
    line[i] = '\0'; // για να γίνει συμβολοσειρά
    printf("Line: %s", line);
}
```



Άλλοι τρόποι με το getchar()

```
int i;  
char line[80];  
i = 0;  
//διαβάζει χαρακτήρες μέχρι να διαβαστεί ο '\n'  
while( (line[i]=getchar( )) != '\n' ) i++;  
//δεν πρέπει να ξεχνάμε το '\0' στο τέλος  
line[i]='\0';  
printf("Diavasa %d caractires\n", i);
```

Αν οι χαρακτήρες είναι περισσότεροι από 80;

```
//Το ίδιο με άλλο τρόπο  
while( (line[i++]=getchar( )) != '\n' );  
line[i] = '\0';
```



Συναρτήσεις του <string.h>

- Στη C υπάρχουν μια σειρά από συναρτήσεις για το χειρισμό των συμβολοσειρών (υπάρχουν στο <string.h>). Η ύπαρξη του μηδενικού χαρακτήρα είναι απαραίτητη για εκτελούνται σωστά αυτές οι ειδικές συναρτήσεις. Κάποιες από αυτές είναι:
 - **strcpy(s1, s2);** Αντιγράφει το s2 στο s1.
 - **strcat(s1, s2);** Συνενώνει τη s2 στο τέλος της s1.
 - **strlen(s1);** Επιστρέφει το μήκος (σε χαρακτήρες) του s1.
 - **strcmp(s1, s2);** Επιστρέφει 0 αν τα s1 και s2 είναι ίδια. Μικρότερο του 0 αν s1<s2. Μεγαλύτερο του 0 αν s1>s2.
 - **strchr(s1, ch);** Επιστρέφει τον δείκτη της πρώτης εμφάνισης του χαρακτήρα ch στη συμβολοσειρά s1.
 - **strstr(s1, s2);** Επιστρέφει τον δείκτη της πρώτης εμφάνισης της συμβολοσειράς s2 στη συμβολοσειρά s1.



Παράδειγματα με τις συναρτήσεις των συμβολοσειρών

```
#include <stdio.h>
#include <string.h>
main () {
    char str1[12] = "Hello";
    char str2[12] = "World";
    char str3[12];
    int len ;
    strcpy(str3, str1);    /* αντιγράφει str1 στο str3 */
    printf("strcpy( str3, str1) : %s\n", str3 );
    strcat( str1, str2);  /* ενώνει str1 και str2 */
    printf("strcat( str1, str2): %s\n", str1 );
    len = strlen(str1);   /* το μήκος του str1 μετά τη συνένωση */
    printf("strlen(str1) : %d\n", len );
}
```

