



Προγραμματισμός Υπολογιστών

Μετατροπή αλφαριθμητικών σε αριθμούς

Νικόλαος Ζ. Ζάχαρης

Καθηγητής

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Η συνάρτηση atoi

```
main.c
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(int argc, char *argv[]) {
5
6      char strValid[15] = "142";
7      char strSomeValid[15] = "127.86 $";
8      char strNotValid[15] = "Hello";
9
10     int num = atoi (strValid);
11     int num1 = atoi (strSomeValid);
12     int num2 = atoi (strNotValid);
13
14     printf("The number (valid) is %d\n", num);
15     printf("The number (some valid) is %d\n", num1);
16     printf("The number (not valid) is %d\n", num2);
17
18     return 0;
19 }
```

```
int atoi(const char *str)
```

```
The number (valid) is 142
The number (some valid) is 127
The number (not valid) is 0
```

```
#include <ctype.h>
int isValid(char *str)
{
    int i, slen = strlen(str);
    if(slen < 1) return 0;
    for(i = 0; i < slen; i++) {
        if( isdigit(str[i]) == 0) {
            return 0;
        }
    }
    return 1;
}
```

Εναλλακτικά

```
#define isNo(x) (x) > '/' && (x) < ':'
```

Η συνάρτηση atoi μετατρέπει ένα αλφαριθμητικό σε ακέραιο αριθμό και επιστρέφει την τιμή του. Σε περίπτωση που το αλφαριθμητικό δεν περιέχει αριθμητικά ψηφία τότε επιστρέφει τη τιμή 0 ή τη ακέραια τιμή μέχρι του σημείου που υπάρχουν αριθμητικά ψηφία.

Η συνάρτηση atoi

```
int main(int argc, char *argv[]) {  
  
    char strValid[15] = "142";  
    char strSomeValid[15] = "127.86 $";  
    char strNotValid[15] = "Hello";  
  
    long num = atoi (strValid);  
    long num1 = atoi (strSomeValid);  
    long num2 = atoi (strNotValid);  
  
    printf("The number (valid) is %ld\n", num);  
    printf("The number (some valid) is %ld\n", num1);  
    printf("The number (not valid) is %ld\n", num2);  
  
    return 0;  
}
```

```
The number (valid) is 142  
The number (some valid) is 127  
The number (not valid) is 0
```

```
long int atoi(const char *str)
```

Η συνάρτηση atoi μετατρέπει ένα αλφαριθμητικό σε μεγάλο ακέραιο αριθμό και επιστρέφει την τιμή του. Σε περίπτωση που το αλφαριθμητικό δεν περιέχει αριθμητικά ψηφία τότε επιστρέφει τη τιμή 0 ή τη ακέραια τιμή μέχρι του σημείου που υπάρχουν αριθμητικά ψηφία.

Η συνάρτηση atof

```
int main(int argc, char *argv[]) {  
  
    char strValid[15] = "142.67";  
    char strSomeValid[15] = "127.86 $";  
    char strNotValid[15] = "Hello";  
  
    double num = atof (strValid);  
    double num1 = atof (strSomeValid);  
    double num2 = atof (strNotValid);  
  
    printf("The number (valid) is %lf\n", num);  
    printf("The number (some valid) is %lf\n", num1);  
    printf("The number (not valid) is %lf\n", num2);  
  
    return 0;  
}
```

```
The number (valid) is 142.670000  
The number (some valid) is 127.860000  
The number (not valid) is 0.000000
```

```
double atof(const char *str)
```

Η συνάρτηση atof μετατρέπει ένα αλφαριθμητικό σε πραγματικό αριθμό και επιστρέφει την τιμή του. Σε περίπτωση που το αλφαριθμητικό δεν περιέχει αριθμητικά ψηφία τότε επιστρέφει τη τιμή 0 ή τη τιμή μέχρι του σημείου που υπάρχουν αριθμητικά ψηφία.

Η συνάρτηση strtod

```
int main(int argc, char *argv[]) {  
  
    char str[15] = "127.86 $";  
    char *ptr;  
    double ret;  
  
    ret = strtod(str, &ptr);  
    printf("The number(double) is %lf\n", ret);  
  
    if(ptr == NULL || strlen(ptr)==0)  
    {  
        printf("The String is a number");  
    }  
    else {  
        printf("String part is %s", ptr);  
    }  
  
    return 0;  
}
```

```
The number(double) is 127.860000  
String part is $
```

```
double strtod(const char *str, char **endptr)
```

Η συνάρτηση strtod μετατρέπει ένα αλφαριθμητικό σε πραγματικό αριθμό (double) και επιστρέφει την τιμή του. Σε περίπτωση που το αλφαριθμητικό δεν περιέχει αριθμητικά ψηφία τότε επιστρέφει τη τιμή 0 ή τη τιμή μέχρι του σημείου που υπάρχουν αριθμητικά ψηφία. Επίσης επιστρέφει μια αναφορά (endptr) στο πρώτο χαρακτήρα που σταμάτησε η μετατροπή του αλφαριθμητικού (str).

Η συνάρτηση strtouf

```
int main(int argc, char *argv[]) {  
  
    char str[15] = "20101.86 $";  
    char *ptr;  
    float ret;  
  
    ret = strtouf(str, &ptr);  
    printf("The number(float) is %f\n", ret);  
  
    if(ptr == NULL || strlen(ptr)==0)  
    {  
        printf("The String is a number");  
    }  
    else {  
        printf("String part is %s", ptr);  
    }  
  
    return 0;  
}
```

```
The number(float) is 20101.859375  
String part is $
```

```
float strtouf(const char *str, char **endptr)
```

Η συνάρτηση strtouf μετατρέπει ένα αλφαριθμητικό σε πραγματικό αριθμό (float) και επιστρέφει την τιμή του. Σε περίπτωση που το αλφαριθμητικό δεν περιέχει αριθμητικά ψηφία τότε επιστρέφει τη τιμή 0 ή τη τιμή μέχρι του σημείου που υπάρχουν αριθμητικά ψηφία. Επίσης επιστρέφει μια αναφορά (endptr) στο πρώτο χαρακτήρα που σταμάτησε η μετατροπή του αλφαριθμητικού (str).

Η συνάρτηση strtol

```
int main(int argc, char *argv[]) {  
  
    char str[15] = "edd.86 $";  
    char *ptr;  
    long int ret;  
  
    ret = strtol(str, &ptr, 16);  
    printf("The number(long int) is %ld\n", ret);  
  
    if(ptr == NULL || strlen(ptr)==0)  
    {  
        printf("The String is a number");  
    }  
    else {  
        printf("String part is %s", ptr);  
    }  
  
    return 0;  
}
```

```
The number(long int) is 3805  
String part is .86 $
```

Η βάση είναι ένας αριθμός από 2 έως 36 και δηλώνει τη μορφή του αριθμού που είναι στο αλφαριθμητικό str. Το 16 είναι το δεκαεξαδικό, οπότε το edd είναι ο αριθμός 3805.

Εάν το str ήταν
char str[15] = "010101.86 \$";
και η βάση 2 τότε το ret θα ήταν 21

```
double strtol(const char *str, char **endptr, int base)
```

Η συνάρτηση strtol μετατρέπει ένα αλφαριθμητικό σε μεγάλο ακέραιο αριθμό και επιστρέφει την τιμή του. Σε περίπτωση που το αλφαριθμητικό δεν περιέχει αριθμητικά ψηφία τότε επιστρέφει τη τιμή 0 ή τη τιμή μέχρι του σημείου που υπάρχουν αριθμητικά ψηφία. Επίσης επιστρέφει μια αναφορά (endptr) στο πρώτο χαρακτήρα που σταμάτησε η μετατροπή του αλφαριθμητικού (str).