Archived:Converting C strings to numbers

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

Overview

The simplest way to convert a string to an integer in Open C is to use `atoi()`. However, `atoi()` does not differentiate between '0' and invalid input and it does not allow specifying which base you are working with. Other analogous functions to `atoi()` are `atof()` and `atol()`. These functions can be used with the variable types "double" and "long".

Functions `strtod()` and `strtol()` offer more control over the conversion process and they will not produce unexpected results on overflow during conversion. These functions make the position of the first unread character in the input string available by assigning it to the second parameter. The function `strtol()` has also a third parameter to indicate the base of the numeral string.

Note: In order to use this code, you need to install Open C plug-in.

This snippet can be self-signed.

MMP file

The following libraries are required:

```
LIBRARY libc.lib
```

Source file

```
#include <stdlib.h> //atoi, atof(), atol(), strtod(), strtol()
#include <stdio.h> //printf

int main(void)
{
  char* int_str = "123"; /* NOTE: atoi will convert e.g. value "xyz" to '0' */
  char* double_str = "123.0";
  char* long_str = "123456";

  int int_result = 0;
  double double_result = 0.0;
  long long_result = 0;

  char* string_to_convert = NULL;
  char* rest_of_the_string = NULL;
  double double_value = 0.0;
  long long_value = 0;

  /* functions atoi(), atof() and atol() */
  int_result = atoi(int_str);
  printf("The string %s as an integer is = %d\n",int_str,int_result);

  double_result = atof(double_str);
  printf("The string %s as a double is = %f\n",double_str,double_result);

  long_result = atol(long_str);
  printf("The string %s as a long is = %ld\n",long_str,long_result);


  /* functions strtod() and strtol() */
  string_to_convert = "123.456RestOfTheString";
  double_value = strtod(string_to_convert, &rest_of_the_string);
  printf("The string = %s, number = %f, rest = %s\n",
        string_to_convert, double_value, rest_of_the_string);

  string_to_convert = "123456789NoMoreNumbers";
  rest_of_the_string = NULL;
  long_value = strtol(string_to_convert, &rest_of_the_string, 10 ); /* use base 10 */
  printf("The string = %s, number = %ld, rest = %s\n",
        string_to_convert, long_value, rest_of_the_string);

  return 0;
}
```

Postconditions

Five different string-to-number conversions have been executed and are displayed as standard output.