

Các hàm chuyển đổi số - chữ

- Mục đích sử dụng
- Mô tả các hàm
- Ví dụ minh họa



Mục đích sử dụng

- Các hàm chuyển đổi số thành chuỗi kí tự và ngược lại nằm trong thư viện `<stdlib.h>`
- Các hàm này cho phép chuyển đổi từ chuỗi kí tự sang số kiểu `int`, `long`, `ulong`, `long long`, `double`
- Có thể chuyển một chuỗi ở hệ base sang giá trị kiểu nguyên
- Có thể chuyển đổi một số ở hệ 10 sang chuỗi ở hệ base

Mô tả các hàm

- Sau đây là các hàm và mô tả chi tiết:

Hàm nguyên mẫu và mô tả
<code>double strtod(const char *s, char **endPtr)</code> Chuyển đổi chuỗi kí tự s thành giá trị double. Nếu không chuyển đổi được giá trị 0 sẽ được trả về. Tham số endPtr trỏ đến kí tự đầu tiên của phần còn lại sau chuyển đổi.
<code>long strtol(const char *s, char **endPtr, int base)</code> Chuyển đổi chuỗi s thành giá trị kiểu long. Hàm trả về 0 nếu không thực hiện chuyển đổi được. Tham số s chứa chuỗi kí tự có giá trị double cần chuyển đổi. Tham số thứ hai endPtr trỏ đến kí tự đầu tiên của phần còn lại của chuỗi gốc sau khi chuyển đổi. Tham số thứ ba là hệ cơ số của giá trị cần chuyển đổi, có thể là hệ 2, 8, 10 hoặc 16...
<code>unsigned long strtoul(const char *s, char **endPtr, int base)</code> Chuyển đổi chuỗi kí tự s ở hệ base sang giá trị kiểu long không dấu. Hàm trả về giá trị của phần chuỗi có thể chuyển đổi ở kiểu long không dấu và endPtr trỏ đến kí tự đầu tiên của phần còn lại sau khi chuyển đổi của chuỗi s. Nếu không chuyển đổi được thì kết quả là 0

Mô tả các hàm

- Sau đây là các hàm và mô tả chi tiết(tiếp):

double atof(const char *s)

Chuyển đổi phần đầu nếu có thể trong chuỗi s thành số thực double. Nếu không chuyển đổi được thì giá trị 0 sẽ được trả về. Mọi kí tự khoảng trắng đầu chuỗi s sẽ được bỏ qua.

int atoi(const char *s)

Chuyển đổi phần đầu nếu có thể trong chuỗi s thành số nguyên kiểu int. Nếu không chuyển đổi được thì giá trị 0 sẽ được trả về. Mọi kí tự khoảng trắng đầu chuỗi s sẽ được bỏ qua.

long atol(const char *s)

Chuyển đổi phần đầu nếu có thể trong chuỗi s thành số nguyên kiểu long. Nếu không chuyển đổi được thì giá trị 0 sẽ được trả về. Mọi kí tự khoảng trắng đầu chuỗi s sẽ được bỏ qua.

long long atoll(const char *s)

Chuyển đổi phần đầu nếu có thể trong chuỗi s thành số nguyên kiểu long long. Nếu không chuyển đổi được thì giá trị 0 sẽ được trả về. Mọi kí tự khoảng trắng đầu chuỗi s sẽ được bỏ qua.

char *_itoa(int n, char *buffer, int base)

Chuyển đổi giá trị số nguyên n thành chuỗi kí tự ở hệ base trong buffer

char *_ltoa(long n, char *buffer, int base)

Chuyển đổi giá trị số nguyên kiểu long n thành chuỗi kí tự ở hệ base trong buffer

Ví dụ minh họa

- Ví dụ sau chuyển số kiểu int và long sang chuỗi:

```
int intNumber = 99999999;  
long longNumber = 999999999L;  
char number1[100];  
char number2[100];  
  
_itoa(intNumber, number1, 2); // chuyển sang hệ 2  
_ltoa(longNumber, number2, 16); // chuyển sang hệ 16  
  
printf("number1 = %s\n", number1);  
printf("number2 = %s\n", number2);
```

```
number1 = 10111110101111000001111111  
number2 = 3b9ac9ff
```

Ví dụ minh họa

- Ví dụ sau chuyển từ chuỗi sang số:

```
char s1[] = "    123456 hello C";  
char s2[] = "**** 12134568";  
char s3[] = "3.141592 hahaha";
```

```
printf("Convert s1 to integer number = %d\n", atoi(s1));  
printf("Convert s2 to integer number = %d\n", atoi(s2));  
printf("Convert s3 to integer number = %d\n", atoi(s3));  
puts("=====");  
printf("Convert s1 to real number = %f\n", atof(s1));  
printf("Convert s2 to real number = %f\n", atof(s2));  
printf("Convert s3 to real number = %f\n", atof(s3));  
puts("=====");  
printf("Convert s1 to long number = %ld\n", atol(s1));  
printf("Convert s2 to long number = %ld\n", atol(s2));  
printf("Convert s3 to long number = %ld\n", atol(s3));
```

Ví dụ minh họa

- Kết quả chuyển đổi:

Convert s1 to integer number = 123456

Convert s2 to integer number = 0

Convert s3 to integer number = 3

=====

Convert s1 to real number = 123456.000000

Convert s2 to real number = 0.000000

Convert s3 to real number = 3.141592

=====

Convert s1 to long number = 123456

Convert s2 to long number = 0

Convert s3 to long number = 3

Ví dụ minh họa

- Chuyển đổi chuỗi thành số:

```
printf("Convert s1 to double = %lf\n", strtod(s1, &endPtr));  
printf("Remain string: %s\n", endPtr);  
printf("Convert s2 to double = %lf\n", strtod(s2, &endPtr));  
printf("Remain string: %s\n", endPtr);  
printf("Convert s3 to double = %lf\n", strtod(s3, &endPtr));  
printf("Remain string: %s\n", endPtr);  
puts("=====");
```

```
Convert s1 to double = 123456.000000  
Remain string:  hello C  
Convert s2 to double = 0.000000  
Remain string:  *** 12134568  
Convert s3 to double = 3.141592  
Remain string:  hahaha  
=====
```


Tiếp theo

Chuẩn định dạng nhập xuất