

字符串转换函数



► 大小写转换

■ `strupr()`

- 原型: `char * strupr(char * str);`
- 功能: 将 `str` 字符串转换成大写字母
- 参数: `str` 将要转换的字符串的首地址
- 返回值: `str` 字符串的首地址

■ `strlwr()`

- 原型: `char * strlwr(char * str);`
- 功能: 将 `str` 字符串转换成小写字母
- 参数: `str` 将要转换的字符串的首地址
- 返回值: `str` 字符串的首地址

► 字符串转数字

■ `atoi()`

- ❑ 原型: `int atoi(const char * nptr);`
- ❑ 功能: `atoi()` 会扫描 `nptr` 字符串, 跳过前面的空白字符, 直到遇到数字或正负号才开始做转换, 而遇到非数字或字符串结束符 (`'\0'`) 才结束转换, 并将结果返回。
- ❑ 参数: `nptr` 待转换的字符串
- ❑ 返回值: 成功转换后的整数, 如果转换不成功则为0

■ 类似的函数:

- ❑ `atof()`: 将一个字符串转换为一个 `double` 类型的数据
- ❑ `atol()`: 将一个字符串转换为一个 `long` 类型的数据
- ❑ `atoll()`: 将一个字符串转换为一个 `long long` 类型的数据

► 字符串转数字

■ strtod()

- ❑ 原型: `double strtod(const char* str, char** endptr);`
- ❑ 功能: 把参数 `str` 指向的字符串转换为一个浮点数 (`double`)。如果 `endptr` 不为空, 则指向转换中最后一个字符后的字符的指针会存储在 `endptr` 引用的位置。
- ❑ 参数: `str` 待转换的字符串, `endptr` 对类型为 `char*` 的对象的引用, 其值由函数设置为 `str` 中数值后的下一个字符的地址。
- ❑ 返回值: 返回转换后的双精度浮点数, 如果没有执行有效的转换, 则返回零 (`0.0`)。

■ 类似的函数:

- ❑ `long int strtol(const char *str, char **endptr, int base);`
- ❑ `strtof()`、`strtoll()`、`strtoul()`、`strtoull()`、`strtold()`

Thanks

