stdio.h（standard input & output）

标准输入输出

stdlib.h （standard library）

标准库头文件。

里面定义了五种类型、一些宏和通用工具函数。类型例如size\_t、wchar\_t、div\_t、ldiv\_t和lldiv\_t； 宏例如EXIT\_FAILURE、EXIT\_SUCCESS、RAND\_MAX和MB\_CUR\_MAX等等； 常用的函数如malloc()、calloc()、realloc()、free()、system()、atoi()、atol()、rand()、srand()、exit()等等。

string.h

借鉴自C 标准库 - <string.h>

头文件定义了一个变量类型、一个宏和各种操作字符数组的函数。

库变量类型

size\_t

这是无符号整数类型，它是 sizeof 关键字的结果。

库宏

NULL

这个宏是一个空指针常量的值。

库函数

strcat（）函数

\*原型：char \*strcat(char \*dest, const char src)

作用：把src所指向的字符串追加到dest 所指向的字符串的结尾。

strcmp（）函数

\*原型：int strcmp(const char \*str1, const char str2)

作用：把 str1 所指向的字符串和 str2 所指向的字符串进行比较。

strcpy（）函数

\*原型：char \*strcpy(char \*dest, const char src)

作用：把 src 所指向的字符串复制到 dest。

strlen（）函数

\*原型：size\_t strlen(const char str)

作用：计算字符串 str 的长度，直到空结束字符，但不包括空结束字符。

memchr（）函数

\*原型：void \*memchr(const void str, int c, size\_t n)

作用：在参数 str 所指向的字符串的前 n 个字节中搜索第一次出现字符 c（一个无符号字符）的位置。

memcmp（）函数

\*原型：int memcmp(const void \*str1, const void str2, size\_t n)

作用：把 str1 和 str2 的前 n 个字节进行比较。

memcpy（）函数

\*原型：void \*memcpy(void \*dest, const void src, size\_t n)

作用：从 src 复制 n 个字符到 dest。

memmove（）函数

\*原型：void \*memmove(void \*dest, const void src, size\_t n)

作用：另一个用于从 src 复制 n 个字符到 dest 的函数。

memset（）函数

\*原型：void \*memset(void str, int c, size\_t n)

作用：复制字符 c（一个无符号字符）到参数 str 所指向的字符串的前 n 个字符。

strncat（）函数

\*原型：char \*strncat(char \*dest, const char src, size\_t n)

作用：把 src 所指向的字符串追加到 dest 所指向的字符串的结尾，直到 n 字符长度为止。

strchr（）函数

\*原型：char \*strchr(const char str, int c)

作用：在参数 str 所指向的字符串中搜索第一次出现字符 c（一个无符号字符）的位置。

strncmp（）函数

\*原型：int strncmp(const char \*str1, const char str2, size\_t n)

作用：把 str1 和 str2 进行比较，最多比较前 n 个字节。

strcoll（）函数

\*原型：int strcoll(const char \*str1, const char str2)

作用：把 str1 和 str2 进行比较，结果取决于 LC\_COLLATE 的位置设置。

strncpy（）函数

\*原型：char \*strncpy(char \*dest, const char src, size\_t n)

作用：把 src 所指向的字符串复制到 dest，最多复制 n 个字符。

strcspn（）函数

\*原型：size\_t strcspn(const char str1, const char str2)

作用：检索字符串 str1 开头连续有几个字符都不含字符串 str2 中的字符。

strerror（）函数

\*原型：char strerror(int errnum)

作用：从内部数组中搜索错误号errnum，并返回一个指向错误消息字符串的指针。

strpbrk（）函数

\*原型：char \*strpbrk(const char \*str1, const char str2)

作用：检索字符串 str1 中第一个匹配字符串 str2 中字符的字符，不包含空结束字符。也就是说，依次检验字符串 str1 中的字符，当被检验字符在字符串 str2 中也包含时，则停止检验，并返回该字符位置。

strrchr（）函数

\*原型：char \*strrchr(const char str, int c)

作用：在参数 str 所指向的字符串中搜索最后一次出现字符 c（一个无符号字符）的位置。

strspn（）函数

\*原型：size\_t strspn(const char \*str1, const char str2)

作用：检索字符串 str1 中第一个不在字符串 str2 中出现的字符下标。

strstr（）函数

\*原型： char \*strstr(const char \*haystack, const char needle)

作用：在字符串 haystack 中查找第一次出现字符串 needle（不包含空结束字符）的位置。

strtok（）函数

\*原型： char \*strtok(char \*str, const char delim)

作用：分解字符串 str 为一组字符串，delim 为分隔符。

strxfrm（）函数

\*原型：size\_t strxfrm(char \*dest, const char src, size\_t n)

作用：根据程序当前的区域选项中的 LC\_COLLATE 来转换字符串 src 的前 n 个字符，并把它们放置在字符串 dest 中。

math.h

头文件中声明了常用的一些数学运算

三角函数

double sin(double);正弦

double cos(double);余弦

double tan(double);正切

反三角函数

double asin (double); 结果介于[-PI/2,PI/2]

double acos (double); 结果介于[0,PI]

double atan (double); 反正切（主值），结果介于[-PI/2,PI/2]

double atan2 (double,double); 反正切（整圆值），结果介于[-PI,PI]

双曲三角函数

double sinh (double);

double cosh (double);

double tanh (double);

指数与对数

double frexp(double value,int \*exp);这是一个将value值拆分成小数部分f和（以2为底的）指数部分exp，并返回小数部分f，即f·2^exp。其中f取值在0.5~1.0范围或者0。

double ldexp(double x,int exp);这个函数刚好跟上面那个frexp函数功能相反，它的返回值是x\*2^exp

double modf(double value,double \*iptr);拆分value值，返回它的小数部分，iptr指向整数部分。

double log (double); 以e为底的对数

double log10 (double);以10为底的对数

double pow(double x,double y);计算x的y次幂

float powf(float x,float y); 功能与pow一致，只是输入与输出皆为单精度浮点数

double exp (double);求取自然数e的幂

double sqrt (double);开平方根

取整

double ceil (double); 取上整，返回不比x小的最小整数

double floor (double); 取下整，返回不比x大的最大整数，即 高斯函数[x]

绝对值

int abs(int i); 求整型的绝对值

double fabs (double);求实型的绝对值

double cabs(struct complex znum);求复数的绝对值

标准化浮点数

double frexp (double f,int \*p); 标准化浮点数，f = x \* 2^p，已知f求x,p (x介于[0.5,1])

double ldexp (double x,int p); 与frexp相反，已知x,p求f

取整与取余

double modf (double,double\*); 将参数的整数部分通过指针回传，返回小数部分

double fmod (double,double); 返回两参数相除的余数

其他

double hypot(double x,double y);已知直角三角形两个直角边长度，求斜边长度

double ldexp(double x,int exponent);计算x\*（2的exponent次幂）

double poly(double x,int degree,double coeffs []);计算多项式

int matherr(struct exception \*e);数学错误计算处理程序

————————————————

版权声明：本文为CSDN博主「张总^\_^」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/u011585791/article/details/115379988