#include<assert.h>//診斷巨集

<http://tw.gitbook.net/c_standard_library/assert_h.html>

#include<ctype.h>//關於字元

<http://tw.gitbook.net/c_standard_library/ctype_h.html>

1 int isalnum(int c) 該函數檢查傳遞的字符是否是字母數字。

2 int isalpha(int c) 該函數是否傳遞的字符是字母。

3 int iscntrl(int c) 該函數是否傳遞的字符是控製字符。

4 int isdigit(int c) 該函數是否傳遞的字符是十進制數字。

5 int isgraph(int c) 該函數是否傳遞的字符的圖形表示，使用的語言環境。

6 int islower(int c) 該函數檢查傳遞的字符是否是小寫字母。

7 int isprint(int c) 該函數檢查傳遞的字符是否是可打印的。

8 int ispunct(int c) 該函數檢查傳遞的字符是否是標點符號。

9 int isspace(int c) 該函數檢查傳遞的字符是否是空白。

9 int isblank(int c)

isblank()和isspace()之間的區別

如果字符為空格，則isspace()僅返回true。換句話說，空白字符是用於分隔一行文本中的單詞的空格字符，並且isblank()用於標識它。

isblank()將空白字符視為製表符(“ \ t”)和空格字符(“”)。

isspace()考慮空格字符：('')-空格，('\ t')-水平製表符，('\ n')-換行符，('\ v')-垂直製表符，('\ f')-提要，( '\ r')-回車

10 int isupper(int c) 該函數檢查傳遞的字符是否是大寫字母。

11 int isxdigit(int c) 該函數檢查傳遞的字符是否是十六進製數字。

|  |  |
| --- | --- |
| **S.N.** | **函數及說明** |
| 1 | [int tolower(int c)](http://tw.gitbook.net/c_standard_library/c_function_tolower.html) 這個函數轉換大寫字母為小寫。 |
| 2 | [int toupper(int c)](http://tw.gitbook.net/c_standard_library/c_function_toupper.html) 這個函數小寫字母轉換為大寫。 |

#include<limits.h> 定義的宏限製的char，int和long類型，如各種變量的值。

<http://tw.gitbook.net/c_standard_library/limits_h.html>

#include<math.h> 各種數學函數和一個宏

<http://tw.gitbook.net/c_standard_library/math_h.html>

|  |  |
| --- | --- |
| **S.N.** | **函數及說明** |
| 1 | [double acos(double x)](http://tw.gitbook.net/c_standard_library/c_function_acos.html) 返回x的反餘弦弧度。 |
| 2 | [double asin(double x)](http://tw.gitbook.net/c_standard_library/c_function_asin.html) 返回x的正弦弧線弧度。 |
| 3 | [double atan(double x)](http://tw.gitbook.net/c_standard_library/c_function_atan.html) 返回x的反正切值，以弧度為單位。 |
| 4 | [double atan2(doubly y, double x)](http://tw.gitbook.net/c_standard_library/c_function_atan2.html) 返回y / x的以弧度為單位的反正切值，根據這兩個值，以確定正確的象限上的標誌。 |
| 5 | [double cos(double x)](http://tw.gitbook.net/c_standard_library/c_function_cos.html) 返回的弧度角x的餘弦值。 |
| 6 | [double cosh(double x)](http://tw.gitbook.net/c_standard_library/c_function_cos.html) 返回x的雙曲餘弦。 |
| 7 | [double sin(double x)](http://tw.gitbook.net/c_standard_library/c_function_sin.html) 返回一個弧度角x的正弦。 |
| 8 | [double sinh(double x)](http://tw.gitbook.net/c_standard_library/c_function_sinh.html) 返回x的雙曲正弦。 |
| 9 | [double tanh(double x)](http://tw.gitbook.net/c_standard_library/c_function_tanh.html) 返回x的雙曲正切。 |
| 10 | [double exp(double x)](http://tw.gitbook.net/c_standard_library/c_function_exp.html) 返回e值的第x次冪。 |
| 11 | [double frexp(double x, int \*exponent)](http://tw.gitbook.net/c_standard_library/c_function_frexp.html) The returned value is the mantissa and the integer yiibaied to by exponent is the exponent. The resultant value is x = mantissa \* 2 ^ exponent. |
| 12 | [double ldexp(double x, int exponent)](http://tw.gitbook.net/c_standard_library/c_function_ldexp.html) Returns x multiplied by 2 raised to the power of exponent. |
| 13 | [double log(double x)](http://tw.gitbook.net/c_standard_library/c_function_log.html) 返回自然對數的x（基準-E對數）。 |
| 14 | [double log10(double x)](http://tw.gitbook.net/c_standard_library/c_function_log10.html) 返回x的常用對數（以10為底）。 |
| 15 | [double modf(double x, double \*integer)](http://tw.gitbook.net/c_standard_library/c_function_modf.html) 返回的值是小數成分（小數點後的部分），並設置整數的整數部分。 |
| 16 | [double pow(double x, double y)](http://tw.gitbook.net/c_standard_library/c_function_pow.html) 返回x的y次方。 |
| 17 | [double sqrt(double x)](http://tw.gitbook.net/c_standard_library/c_function_sqrt.html) 返回x的平方根。 |
| 18 | [double ceil(double x)](http://tw.gitbook.net/c_standard_library/c_function_ceil.html) 返回大於或等於x的最小整數值。 |
| 19 | [double fabs(double x)](http://tw.gitbook.net/c_standard_library/c_function_fabs.html) 返回x的絕對值 |
| 20 | [double floor(double x)](http://tw.gitbook.net/c_standard_library/c_function_floor.html) 返回的最大整數值小於或等於x。 |
| 21 | [double fmod(double x, double y)](http://tw.gitbook.net/c_standard_library/c_function_fmod.html) 返回的x除以y的餘數。 |

#include<stdio.h>定義了三個變量的類型，幾個宏及各種功能進行輸入和輸出。

<http://tw.gitbook.net/c_standard_library/stdio_h.html>

|  |  |
| --- | --- |
| **S.N.** | **變量和說明** |
| 1 | **size\_t** 這是一個無符號整數類型 sizeof關鍵字的結果。 |
| 2 | **FILE** 這是一個對象的類型，適合用於存儲信息的一個文件流。 |
| 3 | **fpos\_t** 這是一個對象類型適用於存儲在一個文件中的任何位置。 |

**stderr, stdin,** and **stdout**  
這些宏的文件類型對應的標準誤差，標準輸入，標準輸出流的指針。

**NULL**  
這個宏是一個空指針常量的值

|  |  |
| --- | --- |
| **S.N.** | **函數及說明** |
| 1 | [void clearerr(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_clearerr.html) Clears the end-of-file and error indicators for the given stream. |
| 2 | [int fclose(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_fclose.html) Closes the stream. All buffers are flushed. |
| 3 | [int feof(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_feof.html) Tests the end-of-file indicator for the given stream. |
| 4 | [int ferror(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_ferror.html) Tests the error indicator for the given stream. |
| 5 | [int fflush(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_fflush.html) Flushes the output buffer of a stream. //清空stdin |
| 6 | [int fgetpos(FILE \*stream, fpos\_t \*pos)](http://tw.gitbook.net/c_standard_library/c_function_fgetpos.html) Gets the current file position of the stream and writes it to pos. |
| 7 | [FILE \*fopen(const char \*filename, const char \*mode)](http://tw.gitbook.net/c_standard_library/c_function_fopen.html) Opens the filename yiibaied to by filename using the given mode. |
| 8 | [size\_t fread(void \*ptr, size\_t size, size\_t nmemb, FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_fread.html) Reads data from the given stream into the array yiibaied to by ptr. |
| 9 | [FILE \*freopen(const char \*filename, const char \*mode, FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_freopen.html) Associates a new filename with the given open stream and same time closing the old file in stream. |
| 10 | [int fseek(FILE \*stream, long int offset, int whence)](http://tw.gitbook.net/c_standard_library/c_function_fseek.html) Sets the file position of the stream to the given offset. The argument *offset* signifies the number of bytes to seek from the given *whence* position. |
| 11 | [int fsetpos(FILE \*stream, const fpos\_t \*pos)](http://tw.gitbook.net/c_standard_library/c_function_fsetpos.html) 設置到給定位置的給定的流文件中的位置。參數*pos* 是由函數的fgetpos 給定的位置。 |
| 12 | [long int ftell(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_ftell.html) 返回給定流的當前文件位置。 |
| 13 | [size\_t fwrite(const void \*ptr, size\_t size, size\_t nmemb, FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_fwrite.html) Writes data from the array yiibaied to by ptr to the given stream. |
| 14 | [int remove(const char \*filename)](http://tw.gitbook.net/c_standard_library/c_function_remove.html) Deletes the given filename so that it is no longer accessible. |
| 15 | [int rename(const char \*old\_filename, const char \*new\_filename)](http://tw.gitbook.net/c_standard_library/c_function_rename.html) Causes the filename referred to by old\_filename to be changed to new\_filename. |
| 16 | [void rewind(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_rewind.html) Sets the file position to the beginning of the file of the given stream. |
| 17 | [void setbuf(FILE \*stream, char \*buffer)](http://tw.gitbook.net/c_standard_library/c_function_setbuf.html) Defines how a stream should be buffered. |
| 18 | [int setvbuf(FILE \*stream, char \*buffer, int mode, size\_t size)](http://tw.gitbook.net/c_standard_library/c_function_setvbuf.html) Another function to define how a stream should be buffered. |
| 19 | [FILE \*tmpfile(void)](http://tw.gitbook.net/c_standard_library/c_function_tmpfile.html) Creates a temporary file in binary update mode (wb+). |
| 20 | [char \*tmpnam(char \*str)](http://tw.gitbook.net/c_standard_library/c_function_tmpnam.html) Generates and returns a valid temporary filename which does not exist. |
| 21 | [int fprintf(FILE \*stream, const char \*format, ...)](http://tw.gitbook.net/c_standard_library/c_function_fprintf.html) Sends formatted output to a stream. |
| 22 | [int printf(const char \*format, ...)](http://tw.gitbook.net/c_standard_library/c_function_printf.html) Sends formatted output to stdout. //定義輸出 |
| 23 | [int sprintf(char \*str, const char \*format, ...)](http://tw.gitbook.net/c_standard_library/c_function_sprintf.html) Sends formatted output to a string. |
| 24 | [int vfprintf(FILE \*stream, const char \*format, va\_list arg)](http://tw.gitbook.net/c_standard_library/c_function_vfprintf.html) Sends formatted output to a stream using an argument list. |
| 25 | [int vprintf(const char \*format, va\_list arg)](http://tw.gitbook.net/c_standard_library/c_function_vprintf.html) Sends formatted output to stdout using an argument list. |
| 26 | [int vsprintf(char \*str, const char \*format, va\_list arg)](http://tw.gitbook.net/c_standard_library/c_function_vsprintf.html) Sends formatted output to a string using an argument list. |
| 27 | [int fscanf(FILE \*stream, const char \*format, ...)](http://tw.gitbook.net/c_standard_library/c_function_fscanf.html) Reads formatted input from a stream. |
| 28 | [int scanf(const char \*format, ...)](http://tw.gitbook.net/c_standard_library/c_function_scanf.html) Reads formatted input from stdin. //定義輸入 |
| 29 | [int sscanf(const char \*str, const char \*format, ...)](http://tw.gitbook.net/c_standard_library/c_function_sscanf.html) Reads formatted input from a string. |
| 30 | [int fgetc(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_fgetc.html) Gets the next character (an unsigned char) from the specified stream and advances the position indicator for the stream. |
| 31 | [char \*fgets(char \*str, int n, FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_fgets.html) Reads a line from the specified stream and stores it into the string yiibaied to by str. It stops when either (n-1) characters are read, the newline character is read, or the end-of-file is reached, whichever comes first. |
| 32 | [int fputc(int char, FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_fputc.html) Writes a character (an unsigned char) specified by the argument char to the specified stream and advances the position indicator for the stream. |
| 33 | [int fputs(const char \*str, FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_fputs.html) Writes a string to the specified stream up to but not including the null character. |
| 34 | [int getc(FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_getc.html) 獲取下一個字符（unsigned char類型）從指定的流，並將流的指針位置。 |
| 35 | [int getchar(void)](http://tw.gitbook.net/c_standard_library/c_function_getchar.html) 從標準輸入獲取一個字符（unsigned char類型）。 |
| 36 | [char \*gets(char \*str)](http://tw.gitbook.net/c_standard_library/c_function_gets.html) 從標準輸入讀取一行，並將其存儲到由str指向的字符串。它時停止讀取換行符或文件結束時達成，以先到為準。 |
| 37 | [int putc(int char, FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_putc.html) 寫入字符到指定的流，並將流的指針位置的參數指定一個字符（unsigned char類型）。 |
| 38 | [int putchar(int char)](http://tw.gitbook.net/c_standard_library/c_function_putchar.html) 寫一個字符（unsigned char類型）的參數指定的字符輸出到stdout。 |
| 39 | [int puts(const char \*str)](http://tw.gitbook.net/c_standard_library/c_function_puts.html) 將一個字符串寫入到stdout，但不包括空字符。一個換行符被追加到輸出。 |
| 40 | [int ungetc(int char, FILE \*stream)](http://tw.gitbook.net/c_standard_library/c_function_ungetc.html) 將字符的字符（unsigned char類型）到指定的流，這是讀取的下一字符。 |
| 41 | [void perror(const char \*str)](http://tw.gitbook.net/c_standard_library/c_function_perror.html) 打印一個描述性的錯誤消息到stderr。首先打印字符串str接著一個冒號，然後是一個空格。 |

#include<stdlib.h>定義四個變量類型，用於執行一般函數的幾個宏和各類函數。

<http://tw.gitbook.net/c_standard_library/stdlib_h.html>

|  |  |
| --- | --- |
| **S.N.** | **功能及說明** |
| 1 | [double atof(const char \*str)](http://tw.gitbook.net/c_standard_library/c_function_atof.html) 轉換參數str指向的字符串到浮點數（double類型）。 |
| 2 | [int atoi(const char \*str)](http://tw.gitbook.net/c_standard_library/c_function_atoi.html) 轉換參數str指向的字符串為整數（int型）。 |
| 3 | [long int atol(const char \*str)](http://tw.gitbook.net/c_standard_library/c_function_atol.html) 轉換字符串參數str指向一個長整型（類型為long int） |
| 4 | [double strtod(const char \*str, char \*\*endptr)](http://tw.gitbook.net/c_standard_library/c_function_strtod.html) 轉換參數str指向的字符串到浮點數（double類型）。 |
| 5 | [long int strtol(const char \*str, char \*\*endptr, int base)](http://tw.gitbook.net/c_standard_library/c_function_strtol.html) 字符串轉換成一個長整型（類型為long int）參數str指向。 |
| 6 | [unsigned long int strtoul(const char \*str, char \*\*endptr, int base)](http://tw.gitbook.net/c_standard_library/c_function_strtoul.html) 轉換字符串參數str指向一個無符號長整型（unsigned long型整數）。 |
| 7 | [void \*calloc(size\_t nitems, size\_t size)](http://tw.gitbook.net/c_standard_library/c_function_calloc.html) 分配請求的內存，並返回一個指向它的指針。 //配置時，預設為0 |
| 8 | [void free(void \*ptr](http://tw.gitbook.net/c_standard_library/c_function_free.html)) calloc，malloc或realloc調用先前分配的回收內存。 //清除配置的記憶體 |
| 9 | [void \*malloc(size\_t size)](http://tw.gitbook.net/c_standard_library/c_function_malloc.html) //配置時，位置初始值為隨機 Allocates the requested memory and returns a yiibaier to it. |
| 10 | [void \*realloc(void \*ptr, size\_t size)](http://tw.gitbook.net/c_standard_library/c_function_realloc.html) //調整配置的大小 嘗試調整的內存塊的大小由ptr指向先前調用malloc或calloc的分配。 |
| 11 | [void abort(void)](http://tw.gitbook.net/c_standard_library/c_function_abort.html) 導致程序異常終止。 |
| 12 | [int atexit(void (\*func)(void))](http://tw.gitbook.net/c_standard_library/c_function_atexit.html) 導致指定的函數功能，當程序正常終止時，被調用。 |
| 13 | [void exit(int status)](http://tw.gitbook.net/c_standard_library/c_function_exit.html) 導致正常程序終止。 |
| 14 | [char \*getenv(const char \*name)](http://tw.gitbook.net/c_standard_library/c_function_getenv.html) name所指向環境字符串的搜索，並返回相關的字符串的值。 |
| 15 | [int system(const char \*string)](http://tw.gitbook.net/c_standard_library/c_function_system.html) 字符串指定的命令被傳遞到主機環境中，要執行的命令處理器。 |
| 16 | [void \*bsearch(const void \*key, const void \*base, size\_t nitems, size\_t size, int (\*compar)(const void \*, const void \*))](http://tw.gitbook.net/c_standard_library/c_function_bsearch.html) 執行二進製搜索。 |
| 17 | [void qsort(void \*base, size\_t nitems, size\_t size, int (\*compar)(const void \*, const void\*))](http://tw.gitbook.net/c_standard_library/c_function_qsort.html) 對數組排序 //快速排序 |
| 18 | [int abs(int x)](http://tw.gitbook.net/c_standard_library/c_function_abs.html) 返回x的絕對值。 |
| 19 | [div\_t div(int numer, int denom)](http://tw.gitbook.net/c_standard_library/c_function_div.html) 數（分子numer）除以分母（分母denom）。 |
| 20 | [long int labs(long int x)](http://tw.gitbook.net/c_standard_library/c_function_labs.html) 返回x的絕對值。 |
| 21 | [ldiv\_t ldiv(long int numer, long int denom)](http://tw.gitbook.net/c_standard_library/c_function_ldiv.html) 數（分母denom）除以分母（分子numer）。 |
| 22 | [int rand(void)](http://tw.gitbook.net/c_standard_library/c_function_rand.html) 返回一個取值範圍為0到RAND\_MAX之間的偽隨機數。 |
| 23 | [void srand(unsigned int seed)](http://tw.gitbook.net/c_standard_library/c_function_srand.html) 這個函數使用rand函數隨機數生成器的種子。 |
| 24 | [int mblen(const char \*str, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_mblen.html) 返回參數str指向一個多字節字符的長度。 |
| 25 | [size\_t mbstowcs(schar\_t \*pwcs, const char \*str, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_mbstowcs.html) 轉換*pwcs*指向的數組參數str所指向的多字節字符的字符串。 |
| 26 | [int mbtowc(whcar\_t \*pwc, const char \*str, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_mbtowc.html) 檢查參數str所指向的多字節字符。 |
| 27 | [size\_t wcstombs(char \*str, const wchar\_t \*pwcs, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_wcstombs.html) 存儲在數組*pwcs*  多字節字符並存入字符串str中的代碼轉換。 |
| 28 | [int wctomb(char \*str, wchar\_t wchar)](http://tw.gitbook.net/c_standard_library/c_function_wctomb.html) 檢查參數*wchar*多字節字符對應的代碼。 |

#include<string.h>定義了一個變量的類型，操作字符數組的一個宏觀和各種功能

<http://tw.gitbook.net/c_standard_library/string_h.html>

|  |  |
| --- | --- |
| **S.N.** | **函數及說明** |
| 1 | [void \*memchr(const void \*str, int c, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_memchr.html) 搜索第一次出現的字符c（unsigned char類型）的前n個字節的字符串參數str指向。 |
| 2 | [int memcmp(const void \*str1, const void \*str2, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_memcmp.html) 比較str1和str2的前n個字節。 |
| 3 | [void \*memcpy(void \*dest, const void \*src, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_memcpy.html) 從src複製n個字符到dest。 |
| 4 | [void \*memmove(void \*dest, const void \*src, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_memmove.html) 另一個函數來複製n個字符到str1。 |
| 5 | [void \*memset(void \*str, int c, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_memset.html) 複製字符c（unsigned char類型）的前n個字符的字符串參數str指向。 |
| 6 | [char \*strcat(char \*dest, const char \*src)](http://tw.gitbook.net/c_standard_library/c_function_strcat.html) 追加src指向的字符串到dest指向的字符串的結束。 |
| 7 | [char \*strncat(char \*dest, const char \*src, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_strncat.html) 追加src指向字符串結尾的字符串指向dest中最多n個字符長。 |
| 8 | [char \*strchr(const char \*str, int c)](http://tw.gitbook.net/c_standard_library/c_function_strchr.html) 搜索第一次出現的字符串中的字符c（unsigned char類型）參數str指向。 |
| 9 | [int strcmp(const char \*str1, const char \*str2)](http://tw.gitbook.net/c_standard_library/c_function_strcmp.html) 比較字符串str1指向指向的字符串str2。 |
| 10 | [int strncmp(const char \*str1, const char \*str2, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_strncmp.html) str1和str2的前n個字節相比。 |
| 11 | [int strcoll(const char \*str1, const char \*str2)](http://tw.gitbook.net/c_standard_library/c_function_strcoll.html) 比較字符串str1的STR2。其結果是依賴於位置LC\_COLLATE設置。 |
| 12 | [char \*strcpy(char \*dest, const char \*src)](http://tw.gitbook.net/c_standard_library/c_function_strcpy.html) 複製字符串src指向到dest。 |
| 13 | [char \*strncpy(char \*dest, const char \*src, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_strncpy.html) 副本最多n個字符的字符串src指向到dest。 |
| 14 | [size\_t strcspn(const char \*str1, const char \*str2)](http://tw.gitbook.net/c_standard_library/c_function_strcspn.html) 計算str1的起始段的長度完全包含在str2中的字符。 |
| 15 | [char \*strerror(int errnum)](http://tw.gitbook.net/c_standard_library/c_function_strerror.html) 搜索一個內部數組的錯誤編號差錯編號，並返回一個指向一個錯誤消息字符串。 |
| 16 | [size\_t strlen(const char \*str)](http://tw.gitbook.net/c_standard_library/c_function_strlen.html) 計算但不包括終止空字符的字符串str的長度。 |
| 17 | [char \*strpbrk(const char \*str1, const char \*str2)](http://tw.gitbook.net/c_standard_library/c_function_strpbrk.html) 查找字符串str1在str2中指定的任何字符相匹配的第一個字符。 |
| 18 | [char \*strrchr(const char \*str, int c)](http://tw.gitbook.net/c_standard_library/c_function_strrchr.html) 參數str指向的字符串中的字符c（unsigned char類型）最後一次出現的搜索。 |
| 19 | [size\_t strspn(const char \*str1, const char \*str2)](http://tw.gitbook.net/c_standard_library/c_function_strspn.html) 計算str1中的初始段完全包含在str2中的字符的長度。 |
| 20 | [char \*strstr(const char \*haystack, const char \*needle)](http://tw.gitbook.net/c_standard_library/c_function_strstr.html) 查找第一次出現的整個字符串針（不包括終止空字符）出現在字符串haystack中。 |
| 21 | [char \*strtok(char \*str, const char \*delim)](http://tw.gitbook.net/c_standard_library/c_function_strtok.html) 中斷字符串str分隔分離成一係列的記號。 |
| 22 | [size\_t strxfrm(char \*dest, const char \*src, size\_t n)](http://tw.gitbook.net/c_standard_library/c_function_strxfrm.html) 轉換前n個字符的字符串src到校報的語言環境和將它們放置字符串dest。 |

#include<time.h>兩個宏和用於操作的日期和時間的各種功能。

<http://tw.gitbook.net/c_standard_library/time_h.html>

|  |  |
| --- | --- |
| **S.N.** | **函數及說明** |
| 1 | [char \*asctime(const struct tm \*timeptr)](http://tw.gitbook.net/c_standard_library/c_function_asctime.html) 將指針返回到一個字符串，它表示結構timeptr 日期和時間。 |
| 2 | [clock\_t clock(void)](http://tw.gitbook.net/c_standard_library/c_function_clock.html) 返回處理器的時鐘使用，因為一個實現定義的時代的開始（通常是程序開頭）。 |
| 3 | [char \*ctime(const time\_t \*timer)](http://tw.gitbook.net/c_standard_library/c_function_ctime.html) 返回一個字符串，代表localtime基於參數定時器。 |
| 4 | [double difftime(time\_t time1, time\_t time2)](http://tw.gitbook.net/c_standard_library/c_function_difftime.html) 返回秒時間1和時間2（時間1時間2）之間的差值。 |
| 5 | [struct tm \*gmtime(const time\_t \*timer)](http://tw.gitbook.net/c_standard_library/c_function_gmtime.html) 定時器的值被分解成結構tm表示以協調通用時間（UTC），也被稱為格林威治標準時間（GMT）。 |
| 6 | [struct tm \*localtime(const time\_t \*timer)](http://tw.gitbook.net/c_standard_library/c_function_localtime.html) 定時器的值被分解成結構tm表示本地時區。 |
| 7 | [time\_t mktime(struct tm \*timeptr)](http://tw.gitbook.net/c_standard_library/c_function_mktime.html) 結構轉換所指出timeptr成一個time\_t值，根據本地時區。 |
| 8 | [size\_t strftime(char \*str, size\_t maxsize, const char \*format, const struct tm \*timeptr)](http://tw.gitbook.net/c_standard_library/c_function_strftime.html) 格式結構timeptr根據定義格式並存儲到str格式規則表示的時間。 |
| 9 | [time\_t time(time\_t \*timer)](http://tw.gitbook.net/c_standard_library/c_function_time.html) 計算當前壓延時間，並將其編碼成time\_t的格式。 |