

Một số hàm toán học trong thư viện math.h

Hàm	Ý nghĩa	Kí hiệu toán học	Ví dụ
sqrt(x)	Căn bậc 2 của x	\sqrt{x}	sqrt(16.0) bằng 4.0
pow(x,y)	x mũ y	x^y	pow(2,3) bằng 8
exp(x)	e mũ x	e^x	exp(1.0) bằng 2.718282
log(x)	logarithm tự nhiên (cơ số e) của x	$\ln x$	log(2.718282) bằng 1.0
log10(x)	logarithm cơ số 10 của x	$\log x$	log10(100) bằng 2
sin(x)	sin của x	$\sin x$	sin(0.0) bằng 0.0
cos(x)	cosin của x	$\cos x$	cos(0.0) bằng 1.0
tan(x)	tang của x	$\tan x$	tan(0.0) bằng 0.0
ceil(x)	phần nguyên già của x, tức là số nguyên nhỏ nhất không nhỏ hơn x	$\lceil x \rceil$	ceil(2.5) bằng 3 ceil(-2.5) bằng -2
floor(x)	phần nguyên non của x, tức là số nguyên lớn nhất không lớn hơn x	$\lfloor x \rfloor$	floor(2.5) bằng 2 floor(-2.5) bằng -3
asin(x)	Tính sin ngược, góc trả về được tính theo rad	Arcsin(x)	
acos(x)	Tính cos ngược, góc trả về được tính theo rad	Arccos(x)	
atan(x)	Tính tan ngược, góc trả về được tính theo rad	Arctan(x)	
fabs(x)	Tính trị tuyệt đối của x (x là giá trị thực)	$ x $	
fmod(x,y) (*)	Chia lấy phần dư của x cho y, trong đó x và y là hai số thực	$x \% y$	fmod(3.5, 2) = 1.5
rint(x) (*)	Làm tròn tới số nguyên gần nhất của giá trị thực x		rint(3.6)= rint(3.2)=

Một số hằng số trong thư viện math.h

M_E	–	giá trị của e = 2.71828.
M_LOG2E	–	logarithm cơ số 2 của e.
M_LOG10E	–	Logarithm cơ số 10 của e.
M_LN2	–	Logarithm cơ số tự nhiên của 2.
M_LN10	–	Logarithm cơ số tự nhiên của 10.
M_PI	--	π .

M_PI_2	--	$\pi/2$.
M_PI_4	--	$\pi/4$.
M_1_PI	--	$1/\pi$.
M_2_PI	--	$2/\pi$.
M_2_SQRTPI	--	$2/\sqrt{\pi}$.
M_SQRT2	--	căn bậc hai của 2.
M_SQRT1_2	--	căn bậc hai của 1/2.

Một số hàm toán học nằm trong thư viện `stdlib.h`

Hàm	Ý nghĩa	Kí hiệu toán học	Ví dụ
abs(x)	Tính trị tuyệt đối của x (x là giá trị nguyên kiểu int)	$ x $	
atof(str)	Chuyển đổi một chuỗi ký tự thành số thực kiểu double		
atoi(str)	Chuyển đổi một chuỗi ký tự thành số nguyên kiểu int		
atol(str)	Chuyển đổi một chuỗi ký tự thành số nguyên kiểu long		
labs(x)	Tính trị tuyệt đối của x (x là giá trị nguyên kiểu long)	$ x $	
strtod(str, NULL, base)	Chuyển đổi một chuỗi ký tự thành số thực kiểu double		strtod ("1234", NULL, 8)=5349
strtol(str, NULL, base)	Chuyển đổi một chuỗi ký tự thành số nguyên kiểu long		strtol ("1234", NULL, 10)=1234
strtoul(str, NULL, base)	Chuyển đổi một chuỗi ký tự thành số nguyên dương kiểu unsigned long		strtoul ("F", NULL, 16)=15
rand()	Sinh số nguyên ngẫu nhiên (từ 0 đến 32767)		
srand(n)	Tạo nhân sinh số ngẫu nhiên, n là số nguyên không dấu		<code>srand((unsigned int)time((time_t *)NULL));</code>
system(str)	Thực hiện một lệnh		<code>system("pause");</code>

Một số hàm trong thư viện `ctype.h`

Hàm	Ý nghĩa	Ví dụ
isalnum (c)	Kiểm tra một ký tự có phải là chữ cái hoặc số hay không (A-Z, a-z, 0-9)	<code>isalnum ('a') → TRUE</code>
isalpha(c)	Kiểm tra một ký tự có phải là chữ cái hay không (A-Z, a-z)	<code>isalpha('3') → FALSE</code>
isdigit(c)	Kiểm tra một ký tự có phải là chữ số hay không (0-9)	
isctrl(c)	Kiểm tra một ký tự có phải là ký tự điều khiển hoặc	

	delete hay không (mã ASCII từ 0-31 and 127)	
isgraph(c)	Kiểm tra một ký tự có phải là ký in được hay không (ngoại trừ ký tự cách trống 32)	
islower (c)	Kiểm tra một chữ cái là chữ thường (a-z).	
isupper (c)	Kiểm tra một chữ cái là chữ hoa (A-Z).	
isprint(c)	Kiểm tra xem có phải ký tự in được hay không (mã ASCII từ 32-126).	
ispunct(c)	Kiểm tra xem ký tự có phải là dấu chấm câu hay không (mã ASCII là 32-47, 58-63, 91-96, 123-126)	
isspace (c)	Kiểm tra ký tự có phải là dấu cách trống hay không	
isxdigit (c)	Kiểm tra ký tự có phải là chữ số trong hệ hexa hay không (0-9, A-F, a-f).	
toupper(c)	Chuyển ký tự chữ cái từ thường thành hoa (nếu không phải chữ cái thì không làm gì cả)	
tolower (c)	Chuyển ký tự chữ cái từ hoa thành thường (nếu không phải chữ cái thì không làm gì cả)	

Một số hàm trong thư viện time.h

Hàm	Ý nghĩa	Ví dụ

Một số hàm trong thư viện stdio.h

Hàm	Ý nghĩa	Ví dụ
fclose(fHandler)	Đóng file đang mở	
feof(fHandler)	Kiểm tra xem đã đến cuối file hay chưa	
fflush(fHandler)	Đổ bộ nhớ đệm của file	fflush(stdin)
getc	Đọc một ký tự từ bàn phím	
getchar	Đọc một ký tự từ bàn phím	
gets	Đọc vào một xâu ký tự từ bàn phím (chấp nhận cả dấu cách trống)	
printf	Hàm xuất dữ liệu	
putchar	In nội dung một ký tự ra màn hình	
puts	In nội dung một xâu ký tự ra màn hình	
scanf	Hàm nhập dữ liệu	
sscanf		
sprintf		

Một số hàm trong thư viện string.h

Hàm	Ý nghĩa	Ví dụ
strcat	Nối hai xâu ký tự lại làm một	char *strcat(char *dest, const char *src);
strncat	Nối hai xâu ký tự lại làm một (chỉ nối n ký tự đầu)	char *strncat(char *dest, const char *src, size_t n);

	của xâu nguồn)	
<u>strlen</u>	Trả về độ dài của một xâu ký tự (số ký tự có trong xâu)	<code>char str[]="Hello!"; printf("%d",strlen(str));</code>
<u>strcmp</u>	So sánh hai xâu ký tự, trả về giá trị < 0 nếu s1 nhỏ hơn s2 0 nếu s1 == s2 > 0 nếu s1 lớn hơn s2	<code>if(strcmp(str,word)==0) printf("They are equal!");</code>
<u>strncmp</u>	So sánh n ký tự đầu của hai xâu (giống strcmp)	<code>strcmp(str,word,3)</code>
<u>strcpy</u>	Copy nội dung của xâu thứ hai vào xâu thứ nhất	<code>strcpy(str,word);</code>
<u>strncpy</u>	Copy n ký tự đầu của xâu thứ hai vào xâu thứ nhất	<code>strcpy(str,word,3);</code>
<u>strchr</u>	Tìm kiếm một ký tự trong xâu. Nếu tìm thấy thì trả về con trỏ trỏ đến vị trí xuất hiện đầu tiên. Ngược lại trả về con trỏ NULL	<code>int character='w'; char *string="red dwarf"; if(strchr(string, character)) puts("Character found");</code>
<u>strrchr</u>	Tìm kiếm một ký tự trong xâu. Nếu tìm thấy thì trả về con trỏ trỏ đến vị trí xuất hiện cuối cùng. Ngược lại trả về con trỏ NULL	<code>int character='w'; char *string="red dwarf"; if(strrchr(string, character)) puts("Character found");</code>
<u>strstr</u>	Tìm kiếm một xâu con trong xâu ký tự ban đầu. Hàm trả về con trỏ tới vị trí tìm được, nếu không tìm thấy thì trả về NULL	<code>char string1[]="red dwarf"; char string2[]="dwarf"; void *pointer; pointer = strstr(string1, string2);</code>
<u>strspn</u>	Tính chiều dài của đoạn đầu tiên trong xâu s mà chứa toàn bộ ký tự trong tập chấp nhận (accept)	<code>size_t strspn(const char *s, const char *accept);</code>
<u>strcspn</u>	Tính chiều dài của đoạn đầu tiên trong xâu s mà không chứa bất kỳ ký tự nào trong tập từ chối (reject)	<code>size_t strcspn(const char *s, const char *reject);</code>
<u>strcasecmp</u>	So sánh hai xâu ký tự, trả về giá trị < 0 nếu s1 nhỏ hơn s2 0 nếu s1 == s2 > 0 nếu s1 lớn hơn s2 Không phân biệt ký tự hoa, thường	<code>if(strcasecmp(str,word)==0) printf("They are equal!");</code>
<u>strncasecmp</u>	So sánh n ký tự đầu của hai xâu (giống strcmp) Không phân biệt ký tự hoa, thường	<code>strncasecmp(str,word,3)</code>