



C++常用查询手册

龙马工作室搜集整理制作



索引

头文件查询

#include<algorithm>-----	3
#include <ctype.h>-----	5
#include < math.h>-----	6
#include < time.h>-----	7
#include < local.h>-----	7
#include < signal.h>-----	7
#include < stdarg.h>-----	8
#include < stdlib.h>-----	8
#include <stdio.h>-----	8
#include < string.h>-----	10
#include < stdlib.h>-----	10

关键字查询

常用字符 ASCII 码查询

头文件查询

#include<algorithm>

STL 通用算法

1. 非修改性序列操作（12 个）

	函数	功能说明
循环	for_each()	对序列中的每个元素执行某操作
查找	find()	在序列中找出某个值的第一次出现的位置
	find_if()	在序列中找出符合某谓词的第一个元素
	find_end()	在序列中找出一子序列的最后一次出现的位置
	find_first_of()	在序列中找出第一次出现指定值集中之值的位置
	adjacent_find()	在序列中找出相邻的一对值
计数	count()	在序列中统计某个值出现的次数
	count_if()	在序列中统计与某谓词匹配的次数
比较	mismatch()	找出两个序列相异的第一个元素
	equal()	两个序列中的对应元素都相同时为真
搜索	search()	在序列中找出一子序列的第一次出现的位置
	search_n()	在序列中找出一值的连续 n 次出现的位置

2. 修改性序列操作（27 个）

	函数	功能说明
复制	copy()	从序列的第一个元素起进行复制
	copy_backward()	从序列的最后一个元素起进行复制
交换	swap()	交换两个元素

	swap_ranges()	交换指定范围的元素
	iter_swap()	交换由迭代器所指的两个元素
变换	transform()	将某操作应用于指定范围的每个元素
替换	replace()	用一个给定值替换一些值
	replace_if()	替换满足谓词的一些元素
	replace_copy()	复制序列时用一给定值替换元素
	replace_copy_if()	复制序列时替换满足谓词的元素
填充	fill()	用一给定值取代所有元素
	fill_n()	用一给定值取代前 n 个元素
生成	generate()	用一操作的结果取代所有元素
	generate_n()	用一操作的结果取代前 n 个元素
删除	remove()	删除具有给定值的元素
	remove_if()	删除满足谓词的元素
	remove_copy()	复制序列时删除具有给定值的元素
	remove_copy_if()	复制序列时删除满足谓词的元素
唯一	unique()	删除相邻的重复元素
	unique_copy()	复制序列时删除相邻的重复元素
反转	reverse()	反转元素的次序
	reverse_copy()	复制序列时反转元素的次序
环移	rotate()	循环移动元素
	rotate_copy()	复制序列时循环移动元素
随机	random_shuffle()	采用均匀分布来随机移动元素
划分	partition()	将满足某谓词的元素都放到前面
	stable_partition()	将满足某谓词的元素都放到前面并维持原顺序

3. 序列排序及相关操作（27 个）

	函数	功能说明
排序	sort()	以很好的平均效率排序
	stable_sort()	排序，并维持相同元素的原有顺序
	partial_sort()	将序列的前一部分排好序
	partial_sort_copy()	复制的同时将序列的前一部分排好序
第 n 个元素	nth_element()	将第 n 各元素放到它的正确位置
二分检索	lower_bound()	找到大于等于某值的第一次出现
	upper_bound()	找到大于某值的第一次出现

	equal_range()	找到（在不破坏顺序的前提下）可插入给定值的最大范围
	binary_search()	在有序序列中确定给定元素是否存在
归并	merge()	归并两个有序序列
	inplace_merge()	归并两个接续的有序序列
有序结构上的集合操作	includes()	一序列为另一序列的子序列时为真
	set_union()	构造两个集合的有序并集
	set_intersection()	构造两个集合的有序交集
	set_difference()	构造两个集合的有序差集
	set_symmetric_difference()	构造两个集合的有序对称差集（并-交）
堆操作	push_heap()	向堆中加入元素
	pop_heap()	从堆中弹出元素
	make_heap()	从序列构造堆
	sort_heap()	给堆排序
最大和最小	min()	两个值中较小的
	max()	两个值中较大的
	min_element()	序列中的最小元素
	max_element()	序列中的最大元素
词典比较	lexicographical_compare()	两个序列按字典序的第一个在前
排列生成器	next_permutation()	按字典序的下一个排列
	prev_permutation()	按字典序的前一个排列

#include <ctype.h>

字符处理

函数	功能说明
isalpha	是否字母
isalnum	是否字母和数字
isctrl	是否控制字符
isdigit	是否数字
isgraph	是否可显示字符（除空格外）
isprint	是否可显示字符（包括空格）
ispunct	是否既不是空格，又不是字母和数字的可显示字符

isspace	是否空格
isupper	是否大写字母
isxdigit	是否 16 进制数字（0—9，A-F）字符
toupper	字符大小写转换函数 转换为大写字母
tolower	转换为小写字母

#include < math.h >

数学函数

ANSI C 标准中各种数学计算函数。

函数	功能说明
abs	求整数的绝对值
acos	反余弦
asin	反正弦
atan	反正切
atan2	反正切 2
cos	余弦
sin	正弦
tan	正切
cosh	双曲函数，双曲余弦
sinh	双曲函数，双曲正弦
tanh	双曲正切
exp	指数函数
frexp	指数分解函数
fdexp	乘积指数函数
log	自然对数
log10	以 10 为底的对数
modf	浮点数分解函数
pow	幂函数
sqrt	平方根函数
fabs	求双精度绝对值
floor	求不大于某数的最大整数
ceil	求不小于某数的最小整数



fmod	求双精度除法的余数，返回为双精度余数
rand	产生-90 到 32767 之间的随机整数

#include < time.h >

日期和时间

包含时间和日期处理函数。

函数	功能说明
asctime	ASCII 码表示的时间
difftime	时间差
mktime	设置时间
clock	处理器时间
ctime	字符串表示的时间
strftime	指定格式的时间

#include < local.h >

地区化

用于处理不同国家的语言差异。

函数	功能说明
setlocale	地区设置
localeconv	数字格式约定查询国家的时间、日期、货币等的格式转换

#include < signal.h >

信号处理

处理在程序执行过程中发生例外的情况。

函数	功能说明
----	------





signal	指定信号处理函数
raise	发送信号

#include < stdarg.h >

可变参数处理

用于实现诸如 printf,scanf 等参数数量可变的函数。

函数	功能说明
va_end	可变参数结束宏
va_start	可变参数开始宏
va_arg	访问下一个可变参数宏

#include < stdlib.h >

动态存储分配

许多 c 编译系统用的是 “molloc.h”，使用时注意查手册。

函数	功能说明
malloc	存储器分配
realloc	重新分配存储器
free	释放存储器
calloc	分配指定数据的内存连续空间

#include <stdio.h>

格式化输入与输出函数

以“流”的方式实现，处理包括文件、控制台等输入输出设备。 、



函数	功能说明
fprintf	格式输出（文件）
fscanf	格式输入（文件）
printf	格式输出（控制台）
scanf	格式输入（控制台）
fclose	关闭文件
fopen	打开文件
feof	文件结尾判断
ferror	文件错误检测
freopen	将已存在的流指针和新文件连接
setbuf	设置磁盘缓冲区
sscanf	从缓冲区中按格式输入
sprintf	格式输出到缓冲区
remove	删除文件
rename	修改文件名称
tmpfile	生成临时文件名称
tmpnam	得到临时文件路径
fgetc	输入一个字符(文件)
fgets	字符串输入(文件)
fputc	字符输出(文件)
fputs	字符串输出(文件)
gets	字符串输入（控制台）
getchar	字符输入（控制台）
getc	字符输入（控制台）
putc	字符输出(控制台)
putchar	字符输出(控制台)
ungetc	字符输出到流的头部
fread	直接流读操作
fwrite	直接流写操作
fgetpos	得到文件位置
fsetpos	文件位置设置
fseek	文件位置移动
ftell	得到文件位置
rewind	文件位置复零位
perror	得到错误提示字符串
clearerr	错误清除
puts	字符串输出(控制台)

#include < string.h >

字符串处理

对字符串进行合并、比较、拷贝等操作。

函数	功能说明
strcpy	串拷贝
strncpy	按长度的串拷贝
strcmp	字符串比较
strncmp	按长度对字符串比较
strcoll	用于非英文字符的字符串比较
strxfrm	字符串转换
memchr	字符查找
memset	字符串设置
strerror	错误字符串映射
strlen	求字符串长度
memmove	目的和源存储区可重叠块拷贝
memcpy	目的和源存储区不可重叠块拷贝

#include < stdlib.h >

实用工具函数

1. 字符串转换函数

函数	功能说明
atoi	字符串转换为整数
atol	字符串转换为长整数
strtod	字符串转换为浮点数
strtol	字符串转换为长整数
strtoul	字符串转换为无符号长整型



2. 伪随机序列产生函数

函数	功能说明
rand	产生随机数
srand	设置随机函数的起动数值

3. 存储管理函数

函数	功能说明
calloc	分配存储器
free	释放存储器
malloc	存储器分配
realloc	重新分配存储器

4. 环境通信

函数	功能说明
abort	中止程序
atexit	退出程序执行，并清除环境变量
exit	退出程序执行
getenv	读取环境参数
system	程序挂起，临时执行一个其他程序
bsearch	搜索和排序工具、二分查找(数据必须已排序)
qsort	快速排序
abs	整数运算函数、求绝对值
div	得到除法运算底商和余数
labs	求长整形底绝对值
ldiv	求长整形除法的商和余数
mblen	多字节字符函数、得到多字节字符的字节数
mbtowc	得到多字节字符的字节数
wctomb	多字节字符转换
mbstowcs	多字节字符的字符串操作、将多字节串转换为整数数组
mcstowbs	将多字节串转换为字符数组





关键字查询

ANSI 标准定义的 C 语言关键字

auto	break	case	const	continue	do	default	double	else
for	enum	float	goto	if	int	long	register	return
struct	switch	char	extern	short	signed	sizeof	static	
typedef	union	unsigned	void	volatile	while			

常用关键字说明

void

声明无返回值或无参数函数，声明无类型指针。

char

字符型类型数据。

int

整型数据，通常为编译器指定的机器字长。

float

单精度浮点型数据。

double

双精度浮点型数据。

short

短整型数据，可修饰的 int。



**long**

长整形数据，可修饰的 int。

signed

有符号数据类型。

unsigned

无符号数据类型。

struct

声明结构体。

union

声明共用体。

enum

声明枚举。

typedef

声明类型别名。

sizeof

用于检测特定类型或特定类型变量的大小。

auto

声明自动变量，由编译器自动分配及释放内存。通常在栈上分配。

static

声明静态变量，修饰函数时，指定函数为文件内部使用。

register

声明寄存器变量。

extern

声明外部变量。

const



与 `volatile` 合称“cv 特性”，指定变量不被当前线程或者进程改变，但有可能被系统、其他线程或进程改变。

volatile

指定变量的值可能会被系统或其他进程/线程改变，强制使编译器每次都要从内存中取得该变量的值。

return

返回特定值。

continue

跳出当前循环，开始下一循环。

break

循环停止或 `switch` 结构停止。

goto

无条件跳转语句。

if

条件语句。

else

条件语句否定分支，常与 `if` 连用。

switch

分支语句。

case

分支语句中的分支标记。

default

分支语句中的优化选择，可选。

for

`for` 循环结构。

do





do 循环结构。

while

while 循环结构。



常用字符 ASCII 码查询

十进制	八进制	十六进制	键盘字符	十进制	八进制	十六进制	键盘字符
0	0	0	NULL	128	200	80	Ç
1	1	1	☺	129	201	81	ü
2	2	2	☹	130	202	82	é
3	3	3	♥	131	203	83	â
4	4	4	♦	132	204	84	ä
5	5	5	♣	133	205	85	à
6	6	6	♠	134	206	86	å
7	7	7	(beep)	135	207	87	ç
8	10	8		136	210	88	ê
9	11	9	(tab)	137	211	89	ë
10	12	A		138	212	8A	è
11	13	B	ð	139	213	8B	ï
12	14	C	♀	140	214	8C	î
13	15	D		141	215	8D	ì
14	16	E	♪	142	216	8E	Ä
15	17	F	☀	143	217	8F	Å
16	20	10	▶	144	220	90	É
17	21	11	◀	145	221	91	æ
18	22	12	↕	146	222	92	Æ
19	23	13	!!	147	223	93	ô
20	24	14	¶	148	224	94	ö
21	25	15	§	149	225	95	ò
22	26	16	—	150	226	96	û
23	27	17	↕	151	227	97	ù
24	30	18	↑	152	230	98	ÿ
25	31	19	↓	153	231	99	Ö
26	32	1A	Σ	154	232	9A	Û

27	33	1B	←	155	233	9B	¢
28	34	1C	└	156	234	9C	£
29	35	1D	↔	157	235	9D	¥
30	36	1E	▲	158	236	9E	₤
31	37	1F	▼	159	237	9F	ƒ
32	40	20		160	240	A0	á
33	41	21	!	161	241	A1	í
34	42	22	"	162	242	A2	ó
35	43	23	#	163	243	A3	ú
36	44	24	\$	164	244	A4	ñ
37	45	25	%	165	245	A5	Ñ
38	46	26	&	166	246	A6	ª
39	47	27	'	167	247	A7	º
40	50	28	(168	250	A8	¿
41	51	29)	169	251	A9	¬
42	52	2A	*	170	252	AA	¬
43	53	2B	+	171	253	AB	½
44	54	2C	,	172	254	AC	¼
45	55	2D	-	173	255	AD	¡
46	56	2E	.	174	256	AE	«
47	57	2F	/	175	257	AF	»
48	60	30	0	176	260	B0	░
49	61	31	1	177	261	B1	▒
50	62	32	2	178	262	B2	▓
51	63	33	3	179	263	B3	▒
52	64	34	4	180	264	B4	▒
53	65	35	5	181	265	B5	▒
54	66	36	6	182	266	B6	▒
55	67	37	7	183	267	B7	▒
56	70	38	8	184	270	B8	▒
57	71	39	9	185	271	B9	▒
58	72	3A	:	186	272	BA	▒
59	73	3B	;	187	273	BB	▒
60	74	3C	<	188	274	BC	▒
61	75	3D	=	189	275	BD	▒
62	76	3E	>	190	276	BE	▒
63	77	3F	?	191	277	BF	▒

64	100	40	@	192	300	C0	└
65	101	41	A	193	301	C1	┐
66	102	42	B	194	302	C2	┘
67	103	43	C	195	303	C3	┙
68	104	44	D	196	304	C4	—
69	105	45	E	197	305	C5	+
70	106	46	F	198	306	C6	└
71	107	47	G	199	307	C7	┐
72	110	48	H	200	310	C8	└
73	111	49	I	201	311	C9	┘
74	112	4A	J	202	312	CA	┐
75	113	4B	K	203	313	CB	┘
76	114	4C	L	204	314	CC	└
77	115	4D	M	205	315	CD	—
78	116	4E	N	206	316	CE	+
79	117	4F	O	207	317	CF	┐
80	120	50	P	208	320	D0	┐
81	121	51	Q	209	321	D1	┘
82	122	52	R	210	322	D2	┘
83	123	53	S	211	323	D3	└
84	124	54	T	212	324	D4	└
85	125	55	U	213	325	D5	┘
86	126	56	V	214	326	D6	┘
87	127	57	W	215	327	D7	+
88	130	58	X	216	330	D8	+
89	131	59	Y	217	331	D9	┘
90	132	5A	Z	218	332	DA	┘
91	133	5B	[219	333	DB	■
92	134	5C	\	220	334	DC	■
93	135	5D]	221	335	DD	■
94	136	5E	^	222	336	DE	■
95	137	5F	_	223	337	DF	■
96	140	60	`	224	340	E0	α
97	141	61	a	225	341	E1	β
98	142	62	b	226	342	E2	Γ
99	143	63	c	227	343	E3	π

100	144	64	d	228	344	E4	Σ
101	145	65	e	229	345	E5	σ
102	146	66	f	230	346	E6	μ
103	147	67	g	231	347	E7	τ
104	150	68	h	232	350	E8	Φ
105	151	69	i	233	351	E9	Θ
106	152	6A	j	234	352	EA	Ω
107	153	6B	k	235	353	EB	δ
108	154	6C	l	236	354	EC	∞
109	155	6D	m	237	355	ED	φ
110	156	6E	n	238	356	EE	ε
111	157	6F	o	239	357	EF	\cap
112	160	70	p	240	360	F0	\equiv
113	161	71	q	241	361	F1	\pm
114	162	72	r	242	362	F2	\geq
115	163	73	s	243	363	F3	\leq
116	164	74	t	244	364	F4	\int
117	165	75	u	245	365	F5	\int
118	166	76	v	246	366	F6	\div
119	167	77	w	247	367	F7	\approx
120	170	78	x	248	370	F8	\circ
121	171	79	y	249	371	F9	\cdot
122	172	7A	z	250	372	FA	\cdot
123	173	7B	{	251	373	FB	$\sqrt{\quad}$
124	174	7C		252	374	FC	n
125	175	7D	}	253	375	FD	2
126	176	7E	~	254	376	FE	■
127	177	7F	\triangle	255	377	FF	

备注：表中 0~127 是标准的。