



C++语言 简明速查手册

刘文杰 赵杰 编

机械工业出版社

C++语言简明速查手册

刘文杰 赵杰 编
李书涛 审



机械工业出版社

本书简明扼要地介绍了C语言及C++语言的函数、语法、全局变量、编译错误和程序设计中经常用到的数表,解释了函数的功能、原型、格式、用法、兼容性及返回值,对语法和常用函数进行了举例说明。本书可根据目录和各种检索方法(关键字、运算符、字母顺序、功能等)快速准确地找到读者所需查看的内容。

该书适合于用C语言和C++语言进行程序设计的工程技术人员和学习C语言的大、中专学生使用。也可作为学习C语言程序设计人员的参考书。

图书在版编目(CIP数据)

C++语言简明速查手册/刘文杰、赵杰编. —北京:机械工业出版社, 1996

ISBN 7-111-05243-9

I. C… I. ①刘…②赵… II. C语言-手册 N. TP312-62

中国版本图书馆CIP数据核字(96)第08572号

出版人:马九荣(北京市百万庄南街1号 邮政编码100037)

责任编辑:钱飒飒 版式设计:张世琴 责任校对:刘文杰

封面设计:方芬 责任印制:卢子祥

三河市宏达印刷厂印刷·新华书店北京发行所发行

1996年11月第1版第1次印刷

787mm×1092mm¹/32·10.875印张235千字

0 001—3 000册

定价20.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

前 言

C 语言及其增强型版本 C++ 语言是一种中级语言。它不仅具有高级语言处理数据的强大功能，而且也具有低级语言直接控制计算机系统的灵活特性。它在程序设计上既支持传统的面向过程设计方法，也支持新的面向对象的设计方法，从而赢得了广大计算机用户的欢迎。许多计算机产业分析家认为：90 年代是 C++ 语言的年代。

C 语言内容丰富，格式多变，仅标准的库函数就有 500 多个，所以用户在工作中不得不经常查阅相关的书籍。我们在讲授 C 语言及 C++ 语言程序设计和进行软件开发中面临的一个棘手的问题就是：对于一些常用的函数、关键字、句法和数表，通常要查阅几本书才能找到。目前已有的几种版本的手册，其篇幅大、源程序冗长，且只介绍库函数，给迅速查阅带来很多不便。由此编者萌发了编写一本既可帮助读者重温 C 语言精华，又可以使读者一书在手，初、中级编程不用愁的简明实用手册的念头。

在编写过程中，我们力求基础内容全面，以实用为主、够用为度，突出重点。本书介绍了库函数、关键字、运算符、常用语法、编译错误及警告和常用数表（ASCII 码及其扩展码表、常用 BIOS 和 DOS 中断表等）。重点是常用函数的说明和使用。本书采用了多种检索方法，函数按字母顺序排列，使查找方便快捷。本书格式精练，节省了大量的篇幅。

本书适合于具有初、中级 C 语言和 C++ 语言编程能力

的工程技术人员使用，也可作为学习 C 语言的大、中专院校学生的参考书。

本书由刘文杰、赵杰编写，刘文杰统稿。本书由李书涛审阅。

由于编者的水平和经验所限，本手册肯定有许多缺点和不足之处，请读者批评指正。

编者 1996 年 4 月

使 用 说 明

一、查阅方法

1. 关键字、运算符、语法可分别由关键字和运算符的检索迅速查到，也可根据第三章、第四章的目录查找。

2. 函数的详细解释可按字母的顺序直接查找，也可以通过功能检索和具有简介函数功能且按字母顺序的检索间接查找。

3. 常用数表由第五章的目录查找。在语法和函数中用到的数表均在相关处指明。

二、条款的解释

【原型】 表示函数的头文件。在使用该函数的源文件首部，必须首先写出头文件。例如：函数 `abort()` 的原型是 `stdlib.h`。在使用该函数的源文件开头写出 `#include "stdlib.h"` 或 `#include <stdlib.h>`。

【举例】 为了节省篇幅，举例中只写出了相关的语句。读者可根据**【原型】**和**【举例】**，按C语言编写风格写出完整程序。例如：

【举例】 `printf ("Calling abort () \n");`
`abort ();`
`return 0; /* 永远不会执行该语句 */`

实际源程序写成：

```
#include "stdlib.h"

int main (void)
```

```
{printf ( "Calling abort () \n");
    abort ();
    return 0; /* 永远不会执行该语句 */ }
```

【移植】 指函数的可移植性。几乎所有的函数在 DOS 系统中都可以使用，而大部分函数在 Windows 系统中也可以使用，因此对 DOS 和 Windows 系统下可使用的函数不再进行说明，而对于其它情况都进行说明。

【返回】 指函数的返回值。在**【格式】**中，函数的类型为 void 时表示函数无返回值。对这类函数，除某些参数有影响的函数，一般不再说明。

【其它】 当函数有两个以上的条款与其它函数相同时，用该条款进行说明。

【参见】 与该函数密切相关和相似的函数用该条款说明。

三 其它

1. 一些具有数学功能的函数有实数和复数两个版本。

例如 abs 函数的解释是：**【原型】** `stdlib.h/math.h, complex.h` **【格式】** `int abs (int i); double abs (complex i);`。实数版本的头文件是 `stdlib.h` 和 `math.h`，函数是 `int abs (int i)`；其它为复数版本。

2. 对一些基本相同的函数，只在其中一个函数中进行了重点说明。

目 录

前言

使用说明

第一章 按字母顺序排列的库函数..... 1

第二章 全局变量..... 228

第三章 C 语言语法及关键字..... 232

1 数据类型 232

1.1 五种基本数据类型 232

1.2 其它数据类型 234

2 运算符及表达式 240

2.1 运算符的特性 240

2.2 常用特殊运算符 240

3 C 语句 241

3.1 ; 空语句 241

3.2 { } 块语句 241

3.3 if 选择语句 241

3.4 switch 开关语句 242

3.5 for 循环语句 242

3.6 while 循环语句 243

3.7 do-while 循环语句 243

3.8 break 转移语句 244

3.9 continue 转移语句 244

3.10 goto 无条件转移语句 244

3.11 return 转移语句 244

3.12 exit 退出语句..... 245

4 C 语言的预处理	245
4.1 #define	245
4.2 #error	245
4.3 #include	245
4.4 #if, #else, #elif 和 #endif	246
4.5 #ifdef 和 #ifndef	246
4.6 #undef	246
4.7 #line	247
第四章 C++ 语言的特性	248
1 C++ 语言的专有特性	248
1.1 变量位置	248
1.2 //注释符号	248
1.3 cin 标准输入流	248
1.4 cout 标准输出流	248
1.5 new 和 delete 用于动态存储分配和释放	249
1.6 inline 内联函数	250
1.7 & 引用	250
1.8 this 指针	251
2 对象和类	251
2.1 class 类	251
2.2 类的对象初始化	253
2.3 private 私有模式	255
2.4 public 公共模式	255
2.5 protected 保护模式	255
2.6 friend 友元函数	255
3 类的继承	256
3.1 派生类的定义	256
3.2 初始化基类成员	258
3.3 virtual 虚函数	258

4	重载	259
4.1	函数重载	259
4.2	operator 运算符重载	259
5	C++ 语言的 I/O 系统	261
5.1	流类及其对象	261
5.2	<< 插入运算符的重载	262
5.3	>> 提取运算符的重载	263
5.4	I/O 的格式化控制	263
6	C++ 文件 I/O	265
6.1	文件的打开和关闭	265
6.2	读写文本文件	266
6.3	读写二进制文件	267
6.4	随机读写	268
	第五章 程序设计的常用数据表	270
表 5-1	ANSI 标准中的数据类型定义	270
表 5-2	特殊输入输出的反斜线代码	270
表 5-3	运算符优先级和结合性	271
表 5-4	I/O 操纵符	273
表 5-5	printf 和 scanf 格式指示符	273
表 5-6	实际颜色值	274
表 5-7	图形操作错误码	275
表 5-8	图形模式	276
表 5-9	调色板颜色表	277
表 5-10	ASCII 字符及扩展字符代码	278
表 5-11	BIOS 中断矢量	281
表 5-12	DOS 中断矢量	288
	第六章 编译错误和警告注解	291
1	一般编译错误	291
2	警告	296

检索	299
检索 1 运算符	299
检索 2 关键字	299
检索 3 按函数功能分类 (包含头文件)	301
检索 4 按函数字母顺序检索 (简述功能)	312
参考文献	335

第一章 按字母顺序排列的库函数

本章按字母顺序较详细地讲解了每个库函数。读者不必从头阅读，需要其中某个函数的信息时，可按字母的顺序直接查找，也可以通过功能检索（见检索 3）和具有简介函数功能且按字母顺序的检索（见检索 4）间接查找。

abort

【功能】 异常终止一个进程。

【原型】 `stdlib.h`。

【格式】 `void abort (void);`

【说明】 立刻结束程序运行，向 `stderr` 写终止信息“Abnormal program termination”。

【返回】 将终止码 3 返回给父进程或 DOS。

【移植】 适用于 UNIX 系统。

【参见】 `assert`, `atexit`, `exit`, `_exit`, `raise`, `signal`, `spawn...`。

【举例】

```
printf ("Calling abort () \n");  
abort ();  
return 0; /* 永远不会执行该语句 */
```

abs

【功能】 求整数绝对值。

【原型】 `stdlib.h/math.h`, `complex.h`。

【格式】 `int abs (int i); double abs (complex i);`

【说明】 本函数计算参数 `i` 的绝对值。

【返回】 实数版本的 `abs` 返回 0 到 32767 间的一个整数，但

当参数值为 -32768 时, 返回 -32768 。复数的 `abs` 返回一个双精度数。

【移植】 实数版本适用于 UNIX 系统。复数版本只适用 C++。

【参见】 `cabs`, `fabs`, `labs`, `complex`。

absread

【功能】 读取磁盘绝对扇区数据。

【原型】 `dos.h`。

【格式】 `int absread (int drive, int nsects, int sectno, void * buffer);`

【说明】 读指定磁盘扇区的内容。它忽略磁盘的逻辑结构, 也不关心文件、FAT 表和目录。`absread` 通过 DOS 中断 `0x25` 读指定的磁盘扇区内容。`drive`=磁盘驱动器号 (`0=A`, `1=B` 等); `nsects`=扇区数; `sectno`=起始扇区号; `buffer`=被读数据的内存地址。最大为 `64K` 或缓冲区的长度, 后者可能比 `64K` 小。

【返回】 成功返回 `0`; 失败返回 `-1`。

【移植】 不能用于 Windows 系统。

【参见】 `abswrite`, `biosdisk`。

abswrite

【功能】 写数据到绝对磁盘扇区。

【原型】 `dos.h`。

【格式】 `int abswrite (int drive, int nsects, int sectno, void * buffer);`

【说明】 此函数是 `absread` 的逆函数, 通过中断 `0x26` 写内容到指定的磁盘扇区。如果使用不正确, 本函数会覆盖文件、目录和 FAT 的内容。

【其它】 见 absread。

access

【功能】 确定文件的存取权限。

【原型】 io.h。

【格式】 int access (const char * filename, int amode);

【说明】 检查 filename 所指定文件是否存在，以及它的可读、可写和可执行权限。amode: 06 检查读/写权限；04 检查读权限；02 检查写权限；01 执行权限（忽略）；00 检查文件是否存在；如果 filename 指向一个目录，access 仅确定该目录是否存在。

【返回】 检验正确返回 0；否则返回 -1。

【移植】 适用于 UNIX 系统。

【参见】 chmod, fstat, stat。

【举例】

```
int file_exists (char * filename)
{
    return (access (filename, 0) == 0);
}

int main (void)
{
    printf ("TEXT.FIL 是否存在?%s", file_exists("
TEXT.FIL") ? "Yes": "No");
    return 0;
}
```

acos

【功能】 计算反余弦值。

【原型】 math.h, complex.h。

【格式】 double acos (double x); complex acos (complex x);

【说明】 计算 x 的反余弦值。传给 acos 的参数必须在 1 和 -

1 之间, 此范围外的值将使 `acos` 返回 `NAN`。复数的反余弦值定义为: $\text{acos}(z) = -i * \log(z + i * \sqrt{1 - z^2})$ 。

【返回】 实数的 `acos` 返回一个范围在 0 到 π 之间的值。

【移植】 与 `abs` 相同。

【参见】 `asin`, `atan`, `atan2`, `complex`, `cos`, `sin`, `tan`, `matherr`。

allocmem

【功能】 分配 DOS 内存段。

【原型】 `dos.h`。

【格式】 `int allocmem (unsigned size, unsigned * seg);`

【说明】 本函数用 DOS 系统调用 `0x48` 来分配自由的内存块。`size` 是字节数, `seg` 为一个指针, 指向将被赋值为新分配块的段地址的地址字。注意: 不能同时使用 `allocmem` 和 `malloc`。

【返回】 成功返回 -1; 若出错将返回可用块的大小。

【参见】 `malloc`, `coreleft`, `freemem`, `setblock`。

【举例】

```
unsigned int size=64, segp;
int stat=allocmen (size, &segp);
if (stat== -1) printf ("分配内存的段地址是%x\n", segp);
```

arc

【功能】 画一圆弧。

【原型】 `graphics.h`。

【格式】 `void far arc (int x, int y, int stangle, int endangle, int radius);`

【说明】 `arc` 用当前绘图颜色, 以 (x, y) 为圆心, `radius` 为半径, 从起始角 `stangle` 到终止角 `endangle` (用度表

示)画一圆弧。若 $\text{stangle}=0$, 且 $\text{endangle}=360$, 则对 arc 的调用就是画一个完整的圆。C++ 中规定 x 轴正向为 0 度, 逆时针方向旋转一周, 顺次为 90 、 180 、 270 和 360 度 (其它有关函数也按此规定, 不再重述)。

【移植】 本函数与 Windows 系统不兼容。

【参见】 circle , ellipse , fillipse , getarccoords , getaspcratio , graphresult , pieslice , sector 。

arg

【功能】 求复平面中的一个复数的弧度。

【原型】 complex.h 。

【格式】 $\text{double arg}(\text{complex } x)$;

【说明】 给出一个复数的弧度值。正实轴弧度为 0 , 负虚轴的弧度为 $\pi/2$ 。如果传给 arg 的参数为复数 0 , 则返回 0 。

【返回】 $\text{atan2}(\text{imag}(x), \text{real}(x))$ 。

【移植】 只适用于 C++ 语言, 不可移植。

【参见】 complex , norm , polar 。

asctime

【功能】 将日期和时间转换为 ASCII 字符串。

【原型】 time.h 。

【格式】 $\text{char} * \text{asctime}(\text{const struct tm} * \text{tblock})$;

【说明】 将在 $* \text{tblock}$ 中作为结构存储的日期和时间值转换为 ASCII 字符串。如: Sun Sep 16 01: 03: 52 1973
 $\backslash \text{n} \backslash 0$ 。

【返回】 返回一个包含日期和时间的字符串指针。

【移植】 适用于 UNIX 系统。

【参见】 ctime, difftime, ftime, gmtime, localtime, mktime, strftime, stime, time, tzset。

【举例】

```
struct tm t;
t.tm_sec=1; /* 秒 */
t.tm_min=30; /* 分 */
t.tm_hour=9; /* 时 */
...
char str [80];
strcpy (str, asctime (&t));
printf ("%s\n", str);
```

asin

【功能】 计算反正弦值。

【原型】 math.h; complex.h。

【格式】 double asin (double x); complex asin (complex x);

【说明】 计算输入值的反正弦值。传给 asin 的参数必须在 -1 和 1 之间, 此范围外的值将使 asin 返回 0。复数的反正弦值定义为: $\text{asin}(x) = -i * \log(i * z + \sqrt{1 - z^2})$ 。

【返回】 实数 asin 返回一个范围在 $-\pi/2$ 到 $\pi/2$ 之间的值。

【移植】 与 abs 相同。

【参见】 matherr, acos, atan, atan2, complex, cos, sin, tan。

assert

【功能】 终止函数。

【原型】 assert.h。

【格式】 void assert (int test);