

全国计算机等级考试二级C语言程序设计 PDF转换可能丢失
图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/134/2021_2022__E5_85_A8_E5_9B_BD_E8_AE_A1_E7_c97_134833.htm

在许多应用软件运行时都带有命令行参数，其实这些命令行参数在C语言编写的程序中也可以实现，灵活地运用命令行参数进行处理可以有效地提高程序的运行效率，收到事半功倍的效果。C语言中有关命令行参数涉及到程序的主函数main(int argc,char *argv[])这样两个参数，其中，int argc表示命令行参数的个数(包括可执行程序名本身)，char *argv[]表示每个参数的具体内容，argv[0]为命令行中可执行程序名本身，argv[1]为命令行中第二个参数的内容，依次类推。如下例输出命令行参数的个数及参数的内容：

```
main (int argc,char *argv[], {int l.  
printf("\n命令行中可执行文件名为： %s",argv[0]). printf("\n总  
共有%d个参数：",argc). l=0. while(argc>=1) {printf( %s  
",argv[l + + ]). argc - - .}} 命令行参数用的最多还是在诸  
如DIR A：等之类带有盘符、路径或文件名这样的命令行中，  
所以说灵活处理这一类参数才能有效地提高程序的运行效果。  
譬如DIR命令，其后可以是盘符，可以是路径，也可以是  
文件名，如何区分这一参数呢?请看下例(此程序模拟DIR命令  
，程序要求在命令行输入一个参数：盘符或路径或文件名，  
若无参数或参数多于一个都将取默认的参数“ *.* ”)。
```

功能：模拟DIR命令
进行处理命令行参数

```
*/ #include #include #include #include int  
j,num=0. char ss[20],path[50],path2[50]. void main (int argc,char
```

```

*argv[]) { struct fblk f. int done. if(argc==2) /*取命令行参数到数
组中*/ strcpy(ss,argv[1]). else strcpy(ss,  *.*  ). /*给数组赋值缺
省参数*/ if (((ss[strlen(ss) - 1]==  \  || ( ( ss[strlen(ss)
- 1]==:)) strcat(ss,  *.*  ). /*若参数为路径或盘符 , 则加上
*.*  */ getcwd(path1,50). /*取当前路径*/ if (chdir(ss)==0) /*
判断参数是否为路径*/ strcat(ss,"\\*.*"). /*若路径未没有带"\" ,
则加上"*.*" */ chdir(path1). /*恢复原来路径*/ strcpy(path2,ss).
for(j=strlen(path2).j>0.j - - )/*提取参数中的路径到path2 */
{if((path2[j]==\\)|| (path2[j]==:)){ path2[j + 1]=\0. goto senull.} }
path2[0]=\0. senull: if(strlen(path2)==0) /* 若给出的参数中没带
路径 , 则取当前路径*/ strcpy(path2,path1). printf("\n**模
拟DIR**\n 命令目录路径%s",path2). done=findfirst(ss, & f,55). /*
查找第一个匹配的文件*/ j=1. while(!done) {if (f.ff_attrib!=0x10)
/* 若文件属性不是目录 */ printf("\n %15s
%20ld",f.ff_name,f.ff_fsize). else printf("\n & 11s ",f.ff_name). num
+ + . j + + . if(j==23) printf("\n - - - - - More (按任
意键继续) - - - - "). getch(). j=0. printf(  \n (目录路径%s)
,path2).} done=findnext( & f). /*查找下一个匹配的文件*/ }
printf(  \n 当前目录中总共有%d个文件.\n  ,num) 100Test 下
载频道开通 , 各类考试题目直接下载。详细请访问
www.100test.com

```