

carekee

Happy Learning

昵称: carekee
园龄: 14年4个月
粉丝: 592
关注: 1
关注成功

<	2023年8月						>
日	一	二	三	四	五	六	
30	31	1	2	3	4	5	
6	7	8	9	10	11	12	
13	14	15	16	17	18	19	
20	21	22	23	24	25	26	
27	28	29	30	31	1	2	
3	4	5	6	7	8	9	

搜索

常用链接

[我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)

积分与排名

积分 - 771199
排名 - 564

随笔档案 (1)

[2012年7月\(1\)](#)

文章分类 (1133)

[AJAX\(14\)](#)
[ASP.NET笔记\(3\)](#)
[ASP.NET转载\(89\)](#)
[BCG\(11\)](#)
[BCGControlBar\(3\)](#)
[C#笔记\(2\)](#)
[C#类\(1\)](#)
[C#转载\(73\)](#)
[C++笔记\(4\)](#)
[C++转载\(178\)](#)
[CSS\(10\)](#)
[CUDA转载\(9\)](#)
[CxImage\(8\)](#)
[C语言\(52\)](#)
[DirectShow\(3\)](#)
[GDI/GDI+\(32\)](#)
[Html\(1\)](#)
[IOCP\(1\)](#)
[JAVA\(3\)](#)
[JavaScript\(109\)](#)

[博客园](#) [首页](#) [新随笔](#) [联系](#) [订阅](#) [RSS](#) [管理](#)

随笔 - 1 文章 - 1218 评论 - 153 阅读 - 506万

一个C语言小程序，有10几个命令和MSDOS一样哦：)

/*调用函数中的scanf前没有提示，其实在主函数中输入时，空格后还可以输入被调用函数中的参数*/

/*这样就避免了很多输出的麻烦，而且看着也比较的舒服:-)例如新建一个文件，就可以直接输入mf a.txt(注意，中间有空格)就可以建立一个文本文件了*/

```
#include<stdio.h>
```

```
#include<dir.h>
```

```
#include<string.h>
```

```
#include<conio.h>
```

```
#include<errno.h>
```

```
#include<dos.h>
```

```
#include<io.h>
```

```
#include<process.h>
```

```
#define PRF printf
```

```
char buffer[MAXPATH], str1[12];
```

```
char file[MAXPATH];
```

/*buffer用来存储工作目录, 定义为全局变量，在用cd..和cd命令有用*/

makefile()/*建立一个新文件*/

```
{
```

```
    int handle;
```

```
    scanf("%s", file);
```

```
    if(strcmp(file, "/?") == 0)
```

```
        PRF(" mf newfilename");
```

```
    else
```

```
    {
```

```
        handle = creatnew(file, 0);
```

```
        if(handle == -1)
```

```
            PRF("%s already exists.\n", file);
```

[jQuery\(5\)](#)

[JSP\(1\)](#)

[Linux/Unix\(22\)](#)

[Linux网络编程\(25\)](#)

[MapXtreme\(48\)](#)

[Matlab\(10\)](#)

[MySQL\(4\)](#)

[OpenCV\(23\)](#)

[Qt\(4\)](#)

[Silverlight\(4\)](#)

[更多](#)

[English Progama Web](#)



[CodeProject](#)

[中文编程论坛](#)



[大家论坛](#)

[中文编程网站](#)



[程序员联合开发网](#)

[w3school](#)

[中文博客](#)



[华夏35度](#)

[闲人阿发伯的业余编程心得](#)

[peterzb](#)

[over140](#)

[阮一峰的网络日志](#)

[China宁](#)

[Gamebaby Rock Sun的博客](#)

[TerryLee's Tech Space](#)

[Jeffrey Zhao](#)

[陈梓瀚](#)

[阅读排行榜](#)



[1. 交通违法记分措施\(5816\)](#)

[最新评论](#)



[1. Re:大数阶乘（转）](#)

棒!!!

--BGM_Bear

[2. Re:在WPF里使用指纹认证（一、安装SDK）](#)

楼主，能分享一下你的demo学习学习吗？谢谢了，求demo
或者您发一份到我邮箱，
2401952147@qq.com

--空空的兜兜

[3. Re:关于“UI线程”（转）](#)

@Outman.Ron.Jim 不会，除非工作线程把CPU资料耗尽了...

--sPhinX

[4. Re:Qt高级——Qt自定义标题栏](#)

博主，源码能发一份吗，给点链接访问不到了，多谢！
2833679472@qq.com

--random_t

[5. Re:Linux学习进阶路线图](#)

else

PRF("%s successfully created.\n", file);

}

}

dir()/*列出当前的工作目录中的文件和目录*/

{

char para[6];

scanf("%s", para);

if(strcmp(para, "?") == 0)

PRF(" dir [/b]/[m]/[p]/[more]/n");

else

if(strcmp(para, "/w") == 0)system("dir/w");

else if(strcmp(para, "/b") == 0)system("dir/b");

else if(strcmp(para, "|more") == 0)system("dir|more");

else if(strcmp(para, "/b|more") == 0)system("dir/b|more");

else if(strcmp(para, "/w|more") == 0)system("dir/w|more");

else if(strcmp(para, "/p") == 0)system("dir/p");

else PRF("'dir' have no '%s' parameter", para);

}

del()/*删除文件*/

{

struct ffbk ffbk; /*定义一个搜索文件的结构体*/

int done, flag=0, i;

scanf("%s", file);

for(i=0; i<strlen(file); i++)

if(file[i] == '?' || file[i] == '*')

{

flag=1;

break;

}

if(strcmp(file, "?") == 0)

PRF(" del filename del *.extendname");

else if(flag)/*删除一特定文件*/

{

done=findfirst(file, &ffb, 0);

while(!done)

{

remove(ffb.ff_name); /*删除找到符合条件的文件*/

done=findnext(&ffb);

}

下面链接都打不开了

--SANJINbky

```
PRF("All files that accord to your conditions were deleted!/n");
}
else
{
    if(remove(file) == 0)
        PRF("Removed %s successfully!/n", file);
    else
        perror("remove"); /*perror()是c语言里的错误处理函数*/
}
}
cd()/*进入选定目录*/
{
    char dirname[MAXPATH];
    scanf("%s", dirname);
    if(strcmp(dirname, "/?") == 0)
        PRF(" cd directoryname");
    else
    {
        strcpy(buffer, dirname); /*将输入的目录名赋给buffer*/
        if(chdir(buffer))/*更改工作目录*/
            perror("chdir");
        getcwd(buffer, MAXPATH); /*用函数chdir()更改目录后，重新给
buffer赋值，使其为当前目录*/
    }
}
cdup()/*返回到上层目录, 有点小问题拉*/
{
    char str_1[MAXPATH], *p;
    int n;
    strcpy(str_1, buffer);
    n=strlen(str_1); /*n存储工作目录buffer字符串的长度*/
    p=str_1+n; /*使指针指向字符串的最后一个字符*/
    while((*p) != '/')
        p--;
    *p='/0'; /*在指针指向str_1的地方重新赋值*/
    strcpy(buffer, str_1);
    if(chdir(buffer))/*更改工作目录*/
        perror("chdir");
    getcwd(buffer, MAXPATH); /*用函数chdir()更改目录后，重新给
buffer赋值，使其为当前目录*/
}
```

```

}
cls()/*清屏*/
{
    clrscr();
}
ver()/*版本信息，纯属好玩:*/
{
    PRF("This programme is created by mabiqiang!\nVersion 1.0 2004-4-14");
}
type()/*显示文件内容*/
{
    FILE *fp;
    struct ftime ft; /*文件被最后修改的时间日期信息*/
    char ch;
    int y; /*定义屏幕上横坐标x和纵坐标y*/
    scanf("%s", file);
    if(strcmp(file, "?") == 0)
        PRF(" type filename(include extend name)");
    else
    {
        if((fp=fopen(file, "r")) == NULL)
            PRF("cannot open %s", file);
        while(!feof(fp))
        {
            ch=fgetc(fp);
            PRF("%c", ch);
            y=wherey(); /*取得当前的纵坐标值*/
            if(y == 25)/*屏幕的高度是25行*/
            {
                printf("\n-----more-----");
                /*当满屏显示后就暂停，直到用户按下任意键后，再继续显示以下的内容*/
                getch();
                clrscr();
            }
        }
        PRF("\n\n This file was modified at ");
        getftime(fileno(fp), &ft); /*取得文件最后修改的时间日期信息*/
        PRF("%u:%u:%u ", ft.ft_hour, ft.ft_min, ft.ft_tsec*2);
    }
}

```

```

        PRF("%u/%u/%u/n", ft.ft_month, ft.ft_day, ft.ft_year+1980);
        fclose(fp);
    }
}

edit()/*编辑一已存在的文件(其实不存在也没关系), 有一小问题就是不能
能够修改, 只能增加数据*/
{
    FILE *fp;
    char ch1, ch2;
    scanf("%s", file);
    if(strcmp(file, "/?") == 0)
        PRF(" edit filename(include extend name)");
    else
    {
        if((fp=fopen(file, "a+")) == NULL)
            PRF("cannot open %s", file);
        while(!feof(fp))
        {
            ch1=fgetc(fp);
            PRF("%c", ch1);
        }
        getchar(); /*清除缓存中的回车字符*/
        ch2=getchar();
        while(ch2 != '$')
        {
            fputc(ch2, fp);
            ch2=getchar();
        }
        fclose(fp);
    }
}

copy()/*复制文件*/
{
    FILE *in, *out;
    char ch, infile[MAXPATH], outfile[MAXPATH];
    scanf("%s", infile);
    scanf("%s", outfile);
    if((in=fopen(infile, "r")) == NULL)
        PRF("cannot open %s", infile);
    if((out=fopen(outfile, "w+")) == NULL)

```

```

        PRF("cannot open %s", outfile);
    while(!feof(in))
        fputc(fgetc(in), out);
    fclose(in);
    fclose(out);
    PRF(" 1 files was copied!");
}
ren()/*改文件名*/
{
    char oldname[MAXPATH], newname[MAXPATH];
    scanf("%s", oldname); /*原文件名*/
    scanf("%s", newname); /*新文件名*/
    if(rename(oldname, newname) == 0)
        PRF("Renamed %s to %s successfully!\n", oldname, newname);
    else
        perror("rename");
}
mdir()/*建立目录*/
{
    int status;
    char dirname[MAXPATH];
    scanf("%s", dirname);
    if(strcmp(dirname, "/?") == 0)
        PRF(" md directoryname/n");
    else
    {
        status = mkdir(dirname);
        (!status)?(PRF("Directory created successfully!\n")):(PRF("Unable
to create directory/n"));
    }
}
rdir()/*删除目录*/
{
    int status;
    char dirname[MAXPATH];
    scanf("%s", dirname);
    if(strcmp(dirname, "/?") == 0)
        PRF(" rd directoryname/n");
    else
    {

```

```

        status = rmdir(dirname);
        (!status)?(PRF("Directory deleted successfully!\n")):
(perror("Unable to delete directory"));
    }
}
date()/*打印当日的日期*/
{
    struct date d;
    getdate(&d);
    PRF(" Today is %d-%d-%d", d.da_year, d.da_mon, d.da_day);
}
time()/*获取当前的时间*/
{
    struct time t;
    gettime(&t);
    PRF(" The current time is:%2d:%02d:%02d", t.ti_hour, t.ti_min,
t.ti_sec);
}
find()/*在当前目录下搜索指定的文件*/
{
    char *p;
    struct fblk fblk;
    int done;
    scanf("%s", file);
    if(strcmp(file, "?") == 0)
        PRF(" find filename");
    else
    {
        PRF("Directory listing of %s:\n", file);
        done=findfirst(file, &fblk, 0);
        while(!done)
        {
            PRF(" %s\n", fblk.ff_name);
            done=findnext(&fblk);
        }
    }
}
change()/*更改驱动盘*/
{
    int disk;

```

```

switch(str1[0])/*将输入字符的第一个字符赋给disk，代表驱动盘*/
{
case 'a':
case 'A':disk=0; break;
case 'c':
case 'C':disk=2; break;
case 'd':
case 'D':disk=3; break;
case 'e':
case 'E':disk=4; break;
case 'f':
case 'F':disk=5; break;
case 'g':
case 'G':disk=6; break;
case 'h':
case 'H':disk=7; break;
}

setdisk(disk); /*设置输入的为当前驱动盘*/

getcwd(buffer, MAXPATH); /*获取当前驱动盘的目录*/
}

call()/*调用子程序并运行*/
{
    int result;
    scanf("%s", file);
    result=spawnl(P_WAIT, file, NULL);
    if(result == -1)/*当调用错误的时候就终止程序*/
    {
        perror("Error from spawnl");
        exit(1);
    }
}

color()/*设置当前文本颜色*/
{
    int txtcol;
    scanf("%d", &txtcol);
    textcolor(txtcol); /*设置文本颜色*/
    clrscr(); /*清屏后才能看到效果*/
}

link()/*将一个文件接到另一个文件的后面*/
{

```



```

FILE *fp1, *fp2;
char ch1, ch2, file1[MAXPATH], file2[MAXPATH];
scanf("%s", file1);
scanf("%s", file2);
if((fp1=fopen(file1, "r")) == NULL)
    PRF("cannot open %s", file1);
/*以a方式打开文件file2时，文件指针已经指向该文件的末尾，可以增加数据*/
if((fp2=fopen(file2, "a")) == NULL)
    PRF("cannot open %s", file2);
while(!feof(fp1))
    fputc(fgetc(fp1), fp2);
fclose(fp1);
fclose(fp2);
PRF(" Link %s to %s successfully!", file1, file2);
}
fc()/*比较两个文件*/
{
    FILE *fp1, *fp2;
    char ch1, ch2, file1[MAXPATH], file2[MAXPATH];
    int flag=0;
    scanf("%s", file1);
    scanf("%s", file2);
    if((fp1=fopen(file1, "r")) == NULL)
        PRF("cannot open %s", file1);
    if((fp2=fopen(file2, "r")) == NULL)
        PRF("cannot open %s", file2);
    while(!feof(fp1) || !feof(fp2))/*当文件都没有结束的时候，依次比较，若有不同，则退出*/
    {
        ch1=fgetc(fp1);
        ch2=fgetc(fp2);
        if(ch1 != ch2)
        {
            flag=1;
            break;
        }
    }
    fclose(fp1);
    fclose(fp2);
}

```

```

        (flag)?PRF("  Two fils are different!");PRF("  Two fils are the same!");
    }
    /*这里可以写自己编写的命令函数*/
    help()/*帮助文件*/
    {
        PRF("  mf  used to creat a new text file/n");
        PRF("  dir  used to show all the files of current directory/n");
        PRF("  del  used to delete a already exist file/n");
        PRF("  cd  used to go to a new directory/n");
        PRF("  cd.. used to go to the directory upon this one/n");
        PRF("  cls  used to clean the screen/n");
        PRF("  ver  used to show the version of this programme/n");
        PRF("  type used to show contents of a already exist file/n");
        PRF("  edit used to edit a file and end with character '$'/n");
        PRF("  copy used to copy a already exist file to another place/n");
        PRF("  ren  used to rename a already exist file/n");
        PRF("  md  used to make a new directory/n");
        PRF("  rd  used to delete a already exist and empty directory/n");
        PRF("  date used to show today's date/n");
        PRF("  time used to show today's time/n");
        PRF("  find used to find a file's in current directory/n");
        PRF("  call used to call a already exist programme and run it/n");
        PRF("  color used to change the text's color/n");
        PRF("  fc  used to compete two files/n");
        PRF("  link used to copy a file's contents to another file/n");
        PRF("All orders can input with file's or directory's name without cls cd..
and ver/n");
        PRF("You can enter like this 'orders /?' for help/n");
    }
    main()
    {
        int flag; /*用于判断输入字符串和命令是否相等, 相等则返回零*/
        PRF("Copyright by MaBiQiang!2004-4-14/nYou can E-mail to
me:mxbgq@sohu.com./n");
        getcwd(buffer, MAXPATH); /*取得当前工作目录*/
        PRF("%s>", buffer);
        while(strcmp(str1, "exit") != 0)
        {
            scanf("%s", str1);
            if(strcmp(str1, "?") == 0 || strcmp(str1, "help") == 0)

```

```

        help();
    else
        if(strcmp(str1, "mf") == 0)flag=1; /*给flag赋不同的值，就执行不同的命令*/
        else if(strcmp(str1, "dir") == 0)flag=2;
        else if(strcmp(str1, "del") == 0)flag=3;
        else if(strcmp(str1, "cd") == 0)flag=4;
        else if(strcmp(str1, "cd..") == 0)flag=5;
        else if(strcmp(str1, "cls") == 0)flag=6;
        else if(strcmp(str1, "ver") == 0)flag=7;
        else if(strcmp(str1, "type") == 0)flag=8;
        else if(strcmp(str1, "edit") == 0)flag=9;
        else if(strcmp(str1, "copy") == 0)flag=10;
        else if(strcmp(str1, "ren") == 0)flag=11;
        else if(strcmp(str1, "md") == 0)flag=12;
        else if(strcmp(str1, "rd") == 0)flag=13;
        else if(strcmp(str1, "date") == 0)flag=14;
        else if(strcmp(str1, "time") == 0)flag=15;
        else if(strcmp(str1, "find") == 0)flag=16;
        else if(strcmp(str1, "A:") == 0 || strcmp(str1, "a:") ==
0)flag=17;
        else if(strcmp(str1, "C:") == 0 || strcmp(str1, "c:") ==
0)flag=18;
        else if(strcmp(str1, "D:") == 0 || strcmp(str1, "d:") ==
0)flag=19;
        else if(strcmp(str1, "E:") == 0 || strcmp(str1, "e:") ==
0)flag=20;
        else if(strcmp(str1, "F:") == 0 || strcmp(str1, "f:") ==
0)flag=21;
        else if(strcmp(str1, "G:") == 0 || strcmp(str1, "g:") ==
0)flag=22;
        else if(strcmp(str1, "H:") == 0 || strcmp(str1, "h:") ==
0)flag=23;
        else if(strcmp(str1, "call") == 0)flag=24;
        else if(strcmp(str1, "color") == 0)flag=25;
        else if(strcmp(str1, "fc") == 0)flag=26;
        else if(strcmp(str1, "link") == 0)flag=27;
        else if(strcmp(str1, "exit") == 0)break;
        /*这里可以加入想要增加的命令，同上*/
    else PRF("%s' is not a effect order of this programme!\n", str1);

```

```

switch(flag)/*根据输入调用命令行*/
{
case 1:makefile(); flag=0; break;
case 2:dir(); flag=0; break;
case 3:del(); flag=0; break;
case 4:cd(); flag=0; break;
case 5:cdup(); flag=0; break;
case 6:cls(); flag=0; break;
case 7:ver(); flag=0; break;
case 8:type(); flag=0; break;
case 9:edit(); flag=0; break;
case 10:copy(); flag=0; break;
case 11:ren(); flag=0; break;
case 12:mdir(); flag=0; break;
case 13:rdir(); flag=0; break;
case 14:date(); flag=0; break;
case 15:time(); flag=0; break;
case 16:find(); flag=0; break;
case 17:
case 18:
case 19:
case 20:
case 21:
case 22:
case 23:change(); flag=0; break;
case 24:call(); flag=0; break;
case 25:color(); flag=0; break;
case 26:fc(); flag=0; break;
case 27:link(); flag=0; break;
/*这里可以加入想要增加的命令的调用函数，同上*/
}
PRF("/n%s>", buffer);
}
}

```

分类: [C语言](#)

好文要顶

关注成功

收藏该文



carekee
粉丝 - 592 关注 - 1

1

推荐

0

反对

支持成功

[刷新评论](#) [刷新页面](#) [返回顶部](#)

B

 自动补全

Powered by .NET 7.0 on Kubernetes