

# 云南大学数学与统计学实验教学中心

## 实验报告

课程名称：操作系统实验	学期：2015~2016 学年上学期	成绩：
指导教师： 彭程	学生姓名：金洋	学号：20131910023
实验名称：实现 dos 环境下的 cd 和 cd.. 功能		
实验编号：一	实验日期： 9 月 18 日	实验学时： 1
学院： 数学与统计学院	专业： 信息与计算科学	年级： 2013 级

### 一、实验目的

1. 实现 dos 环境下的 cd 和 cd.. 功能；

### 二、实验内容

1. 实现 dos 环境下的 cd 和 cd.. 功能；

### 三、实验环境

Windows 操作系统；

Visual C++ 6.0；

### 四、实验过程

1. 实现 dos 环境下的 cd 和 cd.. 功能；

整个程序分为以下几个功能模块：

- ①获得当前目录，返回的地址存入“path”中；
- ②输入模块；
- ③解析指令；
- ④cd 命令

代码及注释如下

```
#define BUFSIZE 100 /*BUFSIZE 是最多能够保存的地址长度*/
#define HISNUM 12
char buf[BUFSIZE];
typedef struct ENV_HISTORY{
    int start;
    int end;
    char his_cmd[HISNUM][100];
} ENV_HISTORY;//定义结构体类型 ENV_HISTORY，并用 typedef 声明其类型名为 ENV_HISTORY

ENV_HISTORY envhis;
#include<stdio.h>
```

```

#include<stdlib.h>
#include<windows.h>
#include<winbase.h>
#include<wincon.h>
#include<tlhelp32.h>
#include<malloc.h>
#include<string.h>
#include<direct.h>

void main()
{
    /*****声明所需的函数*****/
    void add_history(char *inputcmd);//存储历史路径函数
    void cd_cmd(char *route);//更改路径函数
    /*****声明所需的函数*****/

    char c,*input,*arg[2],path[BUFSIZE];
    int input_len=0, is_bg=0,i,j,k;

    DWORD dwRet;//DWORD 一种数据类型, 一个 32 位无符号整数或一段及其相关的地址
    偏移量
    /*****执行命令*****/
    while(1)//使程序不跳出, 持续执行
    {
        /*将指向输入命令的指针数组初始化*/
        for(i=0;i<2;i++)
            arg[i]=NULL;

        /*获得当前目录, 返回的地址存入"path"中*/
        /*BUFSIZE 是最多能够保存的地址长度*/
        dwRet=GetCurrentDirectory(BUFSIZE,path);//获取当前磁盘路径

        if(dwRet==0)
        {
            /*返回当前目录失败, 输出出错信息*/
            printf("GetCurrentDirectory failed (%d)\n",GetLastError());
        }

        else if(dwRet>BUFSIZE)
        {
            /*BUFSIZE 长度小于返回地址长度, 输入需要多少长度*/
            printf("GetCurrentDirectory failed (buffer to small;need %d char)\n",dwRet);
        }
    }
}

```

```

else
{
    /*输出当前目录*/
    printf("%s>",path);
}

/*****输入*****/
input_len=0;

/*将无用字符过滤掉*/
while((c=getchar())!='\t' || c==EOF || c=='\n')
/*    if(c=='\n') 输入为空时，结束本次循环打印提示符
        continue*/;

/*源代码 中上一句并没有注释掉，易造成无用字符为回车时 程序出错，故将上一句注释，
并在 while 语句的判断中加入||c=='\n'*/

/*将命令存入 buf 数组，input_len 记录输入命令长度*/
while(c!='\n')
{
    buf[input_len++]=c;
    c=getchar();
}
buf[input_len++]='\0'; /*加上字符串结束符*/

/*分配动态存储空间，将命令从缓存复制到 input 中*/
input=(char *) malloc(sizeof(char)*(input_len));
strcpy(input,buf);

/*****输入结束*****/

/*****解析指令*****/
for(i=0,j=0,k=0;i<input_len;i++)
{
    if(input[i]==' ' || input[j]=='\0')
    {
        if(j==0)    /*去掉连在一起的多个空格*/
            continue;

        else
        {
            buf[j++]='\0';
            arg[k]=(char*)malloc(sizeof(char)*j);
            strcpy(arg[k++],buf);/*将指令或参数复制到 arg 中*/
            j=0;        /*准备取下一个参数*/
        }
    }
}

```

```

    }
    else
    {
        buf[j++]=input[i];
    }
}

/*****解析指令结束*****/

/*cd 命令*/
if(strcmp(arg[0],"cd")==0)
{
    add_history(input); /*将输入的命令添加到历史命令中*/
    for(i=3,j=0;i<=input_len;i++)/*获取 cd 命令的相关参数*/
        buf[j++]=input[i];
    buf[j]='\0';
    arg[1]=(char*)malloc(sizeof(char)*j);
    strcpy(arg[1],buf);
    cd_cmd(arg[1]); /*显示 cd 命令*/
    free(input);
    continue;
}

/*****cd 命令执行结束*****/
/*dir 命令*/

}

/*****执行命令结束*****/
}

void cd_cmd(char *route)
{
    if(!SetCurrentDirectory(route)) /*设置当前目录，若失败则返回出错信息*/
    {
        printf(TEXT("SetCurrentDirectory failed (%d)\n"),GetLastError());
    }
}

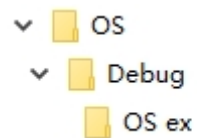
void add_history(char *inputcmd)
{
    envhis.end=(envhis.end+1)%HISNUM;
    /*end 和 start 指向同一数组*/
    if(envhis.end==envhis.start)
    {
        /*start 前移一位*/
        envhis.start=(envhis.start+1)%HISNUM;
    }
}

```

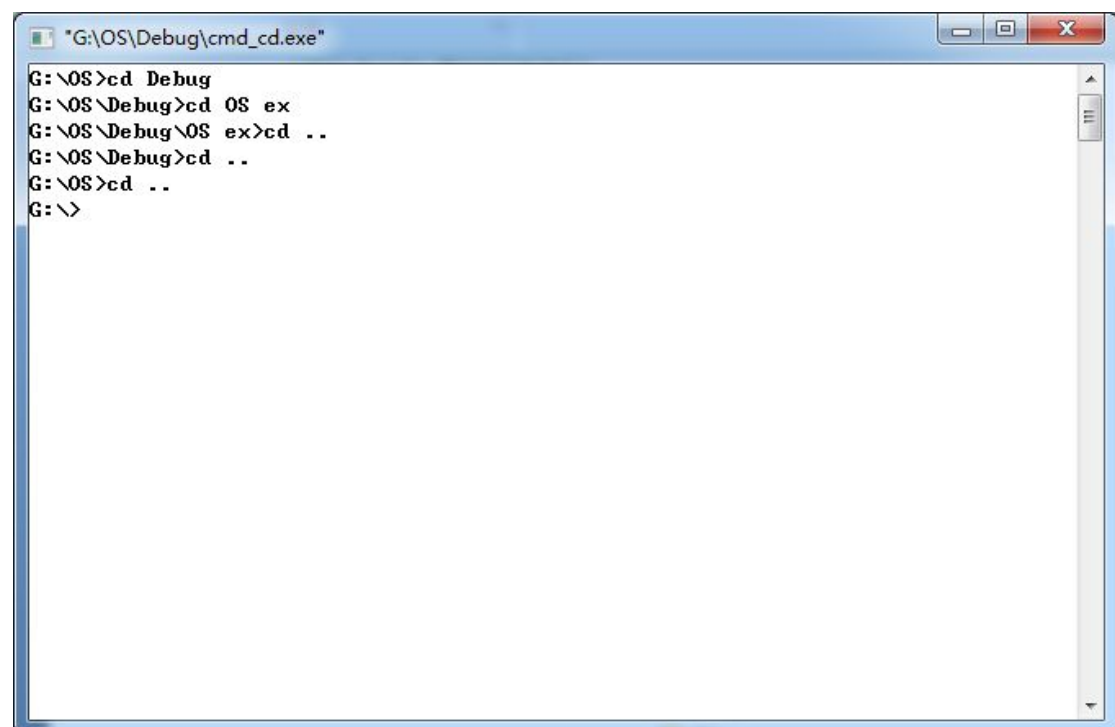
```
}
/*将命令存入 end 指向的数组中*/
strcpy(envhis.his_cmd[envhis.end],inputcmd);
}
```

## 五、实验结果

将程序源文件保存于“OS”文件夹中,OS”文件夹相应子文件如图



得到结果如下



## 六、参考文献

- [1]汤小丹, 梁红兵, 哲凤屏, 汤子瀛. 计算机操作系统[M] (第三版). 西安: 西安电子科技大学出版社, 2007 年 5 月;
- [2]谭浩强著. c 程序设计[M] (第三版). 北京: 清华大学出版社. 2005. 7;

## 七、教师评语