# 实验8 指针实验

## 8.1 实验目的

（1）熟悉文本文件和二进制文件在磁盘中的存储方式。

（2）熟悉掌握流式文件的读写方法。

## 8.2实验内容及要求

8.2.1. 文件类型的程序验证

下面程序是否存在错误？如果存在，原因是什么？如果存在错误，要求在计算机上对这个例子程序进行调试修改，使之能够正确执行。

设有程序：

#include <stdio.h>

int main()

{

short a=0x253f,b=0x7b7d;

char ch;

FILE \*fp1,\*fp2;

fp1=fopen("d:\\abc1.bin","wb+");

fp2=fopen("d:\\abc2.txt","w+");

fwrite(&a,sizeof(short),1,fp1);

fwrite(&b,sizeof(short),1,fp1);

fprintf(fp2,"%hx %hx",a,b);

rewind(fp1);rewind(fp2);

while((ch=fgetc(fp1))!=EOF)

putchar(ch);

putchar('\n');

while((ch=fgetc(fp2))!=EOF)

putchar(ch);

putchar('\n');

fclose(fp1);

fclose(fp2);

return 0;

}

（1）请思考程序的输出结果，然后通过上机运行来加以验证。

（2）将两处sizeof(short)均改为sizeof(char)后，结果有什么不同？为什么？

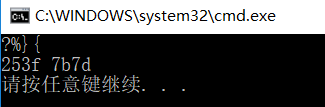
（3）将fprintf(fp2,”%hx%hx”,a,b)改为fprintf(fp2,”%d%d”,a,b)后结果有什么不同。

**解答：**

（1）

wb+二进制文件更新

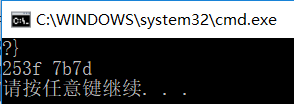
w+ 文本文件更新



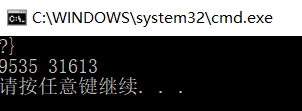
（2）

a为short 两个字节

改为char写入 则低字节写入 高字节舍去



（3）



转换为十进制写入

8.2.2 源程序修改替换

已知将制定文本文件内容在屏幕上显示出来的命令行格式为

type filename

（1）下面源程序中存在什么样的逻辑错误（先观察执行结果）？请对程序进行修改调试，使之能够正确的完成指定任务。

**程序代码：**

#include <stdio.h>

#include <stdlib.h>

int main(int argc,char \*argv[])

{

char ch;

FILE \*fp;

if(argc!=2){

printf("Argument error!\n");

exit(-1);

}

if((fp=fopen(argv[1],"r"))==NULL){ /\*fp指向filename\*/

printf("Can't open %s file!\n",argv[1]);

exit(-1);

}

while(ch=fgetc(fp)!=EOF) /\*从filename中读字符\*/

putchar(ch); /\*向显示器中写字符\*/

fclose(fp); /\*关闭filename\*/

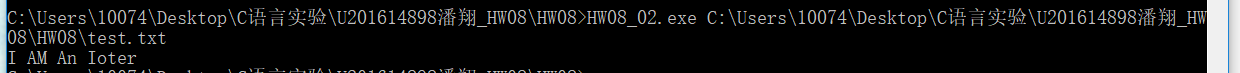
return 0;

}

（2）用输入输出重定向freopen改写上述源程序中的main函数。

**解答：**

（1）

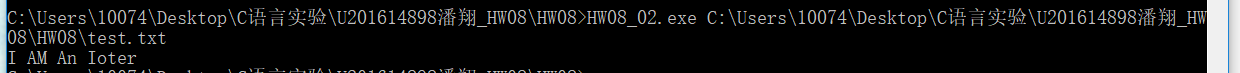


while ((ch = fgetc(fp))!= EOF)

此处需要使用括号来使ch读入

（2）

if ((fp= freopen(argv[1], "r", stdin)) == NULL)



8.2.3 程序设计

编写一个程序，实现以下功能：从键盘输入一行英文句子，将每个单词的首字母换成大写字母，然后输出到一个磁盘文件“test”中保存。

解答

1）算法流程如图8.1所示。



图8-1 编程题1的程序流程图

2）源程序清单

#include <stdio.h>

#include <ctype.h> //用于判断字符类型 字符处理

int main()

{

char s[100];

for (int i = 0; i < 100; ++i)

{

scanf("%c", &s[i]);

if (s[i] == '\n')

{

break;

}

}

s[0] = toupper(s[0]); //将第一个字符变成大写

for (int i = 1; s[i] != '\n'; ++i)

{

if (s[i] == ' ')

{

s[i+1] = toupper(s[i+1]);

i++;

}

}

FILE \*fp;

fp = fopen("test.txt", "w+");

char ch;

for (int i = 0; (ch=s[i]) != '\n'; i++)

{

fputc(ch, fp); //写入文件

putchar(ch); //输出到屏幕

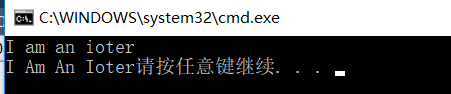
}

fclose(fp);

return 0;

}

3）测试



## 8.5 实验小结

**（1）**使用文件注意指针的初始位置

**（2）**注意文件的关闭

**（3）**不同文件打开方式的掌握