# 8 文件操作实验

## 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. 熟练掌握流式文件的读写方法。

## 8.2 实验内容

**8.2.1 文件类型的程序验证题**

设有程序：

1. #include <stdio.h>
2. int main(void)
3. {
4. short a=0x253f,b=0x7b7d;
5. char ch;
6. FILE \*fp1,\*fp2;
7. fp1=fopen("d:\\abc1.bin","wb+");
8. fp2=fopen("d:\\abc2.txt","w+");
9. fwrite(&a,sizeof(short),1,fp1);
10. fwrite(&b,sizeof(short),1,fp1);
11. fprintf(fp2,"%hx %hx",a,b);
12. rewind(fp1); rewind(fp2);
13. while((ch = fgetc(fp1)) != EOF)
14. putchar(ch);
15. putchar('\n');
17. while((ch = fgetc(fp2)) != EOF)
18. putchar(ch);
19. putchar('\n');
20. fclose(fp1);
21. fclose(fp2);
22. return 0;
23. }
24. 请思考程序的输出结果，然后通过上机运行来加以验证。
25. 将两处sizeof(short)均改为sizeof(char)结果有什么不同，为什么？
26. 将fprintf(fp2,"%hx %hx",a,b) 改为 fprintf(fp2,"%d %d",a,b)结果有什么不同。

**8.2.2 源程序修改替换**

将指定的文本文件内容在屏幕上显示出来，命令行的格式为：

type filename

1. 源程序中存在什么样的逻辑错误（先观察执行结果）？对程序进行修改、调试，使之能够正确完成指定任务。
2. #include<stdio.h>
3. #include<stdlib.h>
4. int main(int argc, char\* argv[])
5. {
6. char ch;
7. FILE \*fp;
8. if(argc!=2){
9. printf("Arguments error!\n");
10. exit(-1);
11. }
12. if((fp=fopen(argv[1],"r"))==NULL){ /\* fp 指向 filename \*/
13. printf("Can't open %s file!\n",argv[1]);
14. exit(-1);
15. }
16. while(ch=fgetc(fp)!=EOF) /\* 从filename中读字符 \*/
17. putchar(ch); /\* 向显示器中写字符 \*/
18. fclose(fp); /\* 关闭filename \*/
19. return 0;
20. }
21. 用输入输出重定向freopen改写main函数。

**8.2.3 程序设计**

1. 编写一个程序replace，采用命令行方式，用给定的字符串替换指定文件中的目标字符串，并显示输出替换的个数。例如，命令行：

replace filename.txt you they

1. 从键盘输入10个单精度浮点数，以二进制形式存入文件float.dat中。再从文件中读出这10个单精度浮点数显示在屏幕上。之后要求将float.dat中的单精度浮点数按字节读出来，观察写入文件的浮点数字节数据是不是和计算机内存中表示的浮点数字节数据一致。