**TEST – 13 文 件**

**测试 - 试题/答案**

【测试时间90分，自行批阅，不计分 】

**学习和解题要点**

一、C 语言的文件基本概念

1. 把内存变量的数据写到磁盘成为文件，也可以从磁盘文件中把数据读入内存。

main()

{ int i,n;

打开文件

n

n

文件名

for(i=1;i<=100;i++){

scanf(“%d”,&n);

把 n 写到文件去

}

关闭文件

}

main()

{ int n;

打开文件

while(文件没结束）{

把文件中的一个数据读到 n

printf(“%d”,n);

}

关闭文件

}

2. 文件类型指针

在 stdio.h 中定义类型名 FILE 为文件类型（一定大写），可以用它定义文件类型的指针变量名。如：fp; 可用 fp 来指向一个磁盘文件。

FILE \*fp;

3. 文件的打开函数

fp = fopen（“文件名”，“使用方式”）；

文件打开成功，返回指针，fp 即指向文件名；否则返回 NULL 即 0。其中：

**文件名：**是操作系统可管理的文件名，还可以加路径。如果是把文件名放在字符数组或字符串指针中，则不加“ ”。

如：fp=fopen(“c:\mao\mao1.dat”, “使用方式”）；

或：char str[20]={“mao1.dat”};

char \*strp=“c:\\mao\\mao1.dat”;

则：fp=fopen(str, “使用方式”）；

fp=fopen(strp, “使用方式”）；

**使用方式：（文本文件）**

“w” (只写）建立一个新文件，如磁盘上已有同名文件，则被重新复盖。

“r” (只读）读一个已有的文件，如磁盘上没有该文件，则打开出错。

“a” (追加）打开一个已有的文件，在该文件尾追加记录。

“w+” (读写）与“w”一样，只不过还可以马上读。

“r+” (读写）与“r”一样，只不过还可以改写其中数据。

“a+” (读写）与“a”一样添加，也可以读。

每一个加上 b 变成 “wb” “rb” “ab” “wb+” “rb+” “ab+” 则对付二进制文件。

4. 文本文件和二进制文件

把同一数据以二种形式写到二个文件上，一种为文本文件，一种为二进制文件。

如 int i=123; 在内存中为二字节二进制：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |

以文本文件形式写到文件上，则用三字节，分别放‘1’字符的ASCII、‘2’字符的ASCII和‘3’字符的ASCII。当然，当再从文件读回到整型内存变量时，仍然变回到二字节的二进制形式。

**文本文件：** ASCII 97 98 99

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |  |  |

以二进制文件形式写到文件上，则仍用二个字节，且以变量在内存中的二进制形式原本的写到文件中。

**二进制文件：**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |  |  |  |  |  |  |  |  |  |  |

5. 文件的关闭

不用的文件要及时关闭，即断开文件指针与磁盘文件的联系。

fclose(fp);

6. 在对文件进行读（写）时，有一个无形的“指示”(实际上就是文件指针)在文件记录上游动，当文件打开时它在第一个记录，每当读一个记录向尾走动一次，有一个判断是否读到文件尾的函数feof(fp)，当碰到文件结束，函数返回1，在文件头或中间，函数返回0。

另外，C 语言在用ch=fgetc(fp)，一个字符一个字符读文本文件时，还可用是否读到EOF即（-1）来判断文本文件是否结束。

C 语言把键盘当成输入用的文本文件，并认为始终以“r”方式打开，不必有文件指针， if(feof()),输入时以Ctrl-z作为文件结束。

同样，C 语言把显示器当成输出用的文本文件，并认为始终以“w”方式打开，不必有文件指针。

二、文本文件的读写

1. 字符读写函数 fgetc(), fputc()

格式：ch=fgetc(fp); 从文件fp中读入一个字符到ch。

fputc(ch,fp); 把ch中的字符写到文件fp。

比较 ch=getchar(); putchar(ch);

2. 字符串读写函数 fgets(), fputs()

格式： fgets(str,n,fp); 从文件fp中读入n-1个字符到str中，若不到n-1个

字符就遇到换行符或文件结束，则有多少就读多少。

fputs(str,fp); 把str中的字符串写到文件fp。

比较 gets(str); puts(str);

3. 格式读写函数 fscanf(), fprintf()

格式：fscanf(fp,“……”,&……);

fprintf(fp,“…………”,………);

比较 scanf(“……”,&……);

printf(“…………”,………);

三、二进制文件的读写

1. 字节读写函数 fgetc(), fputc()

fgetc(), fputc() 可以用来对二进制文件一个字节一个字节的读写，而不管字节中是什 么内容，反正是二进制。

2. fread(), fwrite()

格式：fread(buffer,size,count,fp);

fwrite(buffer,size,count,fp);

buffer 存放数据的起始地址

size 读写的字节数

count 对字节数的重复数

3. getw(), putw()

读写一个字（word 计算机字长)

四、文件的定位

1. rewind(fp) 把“游标”再从头开始。

2. fseek(fp,len,flag); len 位移量，负数为向反方向。

flag 起始点，0 文件头起，1 当前位置起，2 文件尾起。

3. ftell(fp) 测试“游标”当前位置

i=ftell(fp);

五、文件操作检测

1. ferror(fp) 文件操作有错返回非 0，否则为 0。

1. clearerr(fp) 调用后，使ferror(fp)返回 0。

五、文件处理例子

1. 例1. 把键盘输入的整数分别建成文本文件和二进制文件。

#include <stdio.h>

#include <io.h>

void main()

{

FILE \*fp1,\*fp2;

int i;

if((fp1=fopen("data1.txt","w"))==NULL)

{ printf("打不开文件data1.txt\n"); return;}

if((fp2=fopen("data2.txt","wb"))==NULL)

{ printf("打不开文件data2.txt\n"); return;}

scanf("%d",&i);

while(i!=0){

fprintf(fp1,"%d ",i);

fwrite(&i,2,1,fp2);

scanf("%d",&i);

}

return;

}

2. 例2. 把一个文件拷贝成另一个文件。

#include <stdio.h>

void main()

{

FILE \*fp;

void copy\_f(char \*f\_1,char \*f\_2);

char f\_name1[13],f\_name2[13];

printf("输入源文件名: ");

scanf("%s",f\_name1);

printf("输入目标文件名: ");

scanf("%s",f\_name2);

copy\_f(f\_name1,f\_name2);

if((fp=fopen(f\_name2,"r"))==NULL)

{ printf("新文件打不开! \n"); return;}

printf("读入的新文件内容如下:\n");

while(!feof(fp))

putchar(fgetc(fp));

printf("\n");

fclose(fp);

return;

}

void copy\_f(char \*f\_1,char \*f\_2)

{

FILE \*fp1,\*fp2;

if((fp1=fopen(f\_1,"r"))==NULL)

{ printf("打不开被拷贝的文件\n"); return;}

if((fp2=fopen(f\_2,"w"))==NULL)

{ printf("打不开新产生的文件\n"); return;}

while(!feof(fp1))

fputc(fgetc(fp1),fp2);

fclose(fp1); fclose(fp2);

return;

}

**练习题**

1. 单项选择题
2. C语言中，文件由（ ）

A. 字符（字节）序列组成 B. 记录组成

C. 数据行组成 D. 数据块组成

1. 若文件型指针fp中指向某文件的末尾，则函数feof(fp)的返回值是（ ）。

A. 0 B. -1 C. 非零值 D. NULL

1. 状下列语句中，将c定义为文件型指针的是（ ）。

A. FILE c； B. FILE \*c； C. file c； D. file \*c；

1. 以读写方式打开一个已有的文本文件file1，并且FILE \*fp，下面fopen函数正确的调用方式是（ ）。

A. fp=fopen(“file1”,”r”) B. fp=fopen(“file1”,”r+”)

C. fp=fopen(“file1”,”rb”) D. fp=fopen(“file1”,”w”)

1. 标准库函数fputs(p1,p2)的功能是（ ）。

A.从p1指向的文件中读一个字符串存入p2指向的内存

B.从p2指向的文件中读一个字符串存入p1指向的内存

C.从p1指向的内存中读一个字符串写到p2指向的文件中

D.从p2指向的内存中读一个字符串写到p1指向的文件中

二、填充题

1. C语言中调用 函数打开文件，调用 函数关闭文件。
2. feof函数可用于 文件和 文件，它用来判断即将读入的是否为 ，若是，函数值为 。
3. 若ch为字符变量，fp为文本文件指针，从fp所指文件中读入一个字符时，可用的二种不同的输入语句是 和 。把一个字符输出到fp所指文件中的二种不同的输出语句是 和 。
4. sp=fgets(str,n,fp)；函数调用语句从 指向的文件输入 个字符，并把它们放到字符数组str中，sp得到 的地址。而 函数的作用是向指定的文件输出一个字符串，输出成功函数值为 。
5. “FILE \*p”的作用是定义了一个 ，其中的“FILE”是在 头文件中定义的。
6. 在对文件进行操作的过程中，若要求文件的位置回到文件的开头，应当调用的函数是

函数。

1. 下面的程序用来统计文件中字符的个数，请填空。

#include <stdio.h>

main()

{ FILE \*fp;

long num=0;

if(fp=fopen(“fname.dat”,”r”)==NULL)

{ printf(“Can’t open file!\n”); exit(0); }

while { fgetc(fp); num++; }

printf(“num=%d\n”,num);

fclose(fp);

}

1. 下面程序完成从磁盘文件stu.dat中读取N个学生的姓名、学号、成绩后在屏幕上显示输出。请将程序补充完整。

#include <stdio.h>

#define N 100

struct student

{ char name[20];

int no;

int score;

} stud[N];

main()

{ FILE \*fp;

int i;

if

{ printf(“Can not open file!\n”); exit(0); }

;

for(i=0;i<N;i++)

printf(“name:%s NO:%d score:%d\n”,stud[i].name,stud[i].no,stud[i].score);

fclose(fp) ;

}

三、程序分析题

1. 读程序，指出程序所完成的功能。

#include <stdio.h>

main()

{ int ch1,ch2;

while((ch1=getchar())!=EOF)

if (ch1>=’a’ && ch1<=’z’)

{ ch2=ch1-31; putchar(ch2); }

else putchar(ch1);

}

1. 读程序，写功能。

#include <stdio.h>

main()

{ FILE \*fp;

int b=0;

char ch;

if((fp=fopen(“fname.txt”,”r”))==NULL)

{ printf(“Can not open file!\n”); exit(0); }

while(!feof(fp))

{ ch=fgetc(fp); if(ch==’ ‘) b+=1; }

printf(“b=%d\n”,b);

fclose(fp);

}

1. 读程序，写功能。

#include “stdio.h”

main()

{ FILE \*f1, \*f2;

int k;

if((f1=fopen(“c:\tc\p1.c”,”r”))==NULL)

{ printf(“Can not open file!\n”);

exit(0);

}

if((f2=fopen(“a:\p1.c”,”w”))==NULL)

{ printf(“Can not open file!\n”);

exit(0);

}

for(k=1;k<=1000;k++)

{ if(feof(f1) break;

fputc(fgetc(f1),f2);

}

fclose(f1); fclose(f2);

}

1. 假定在当前盘当前目录下有2个文本文件，其名称和内容如下：

文件名 a1.txt a2.txt

内容 121314# 252627#

写出下列程序运行后的输出。

#include “stdio.h”

main()

{ FILE \*fp;

void fc(FILE \*fp1);

if((fp=fopen(“a1.txt”,”r”))==NULL)

{ printf(“Can not open file!\n”); exit(1); }

else { fc(fp); fclose(fp); }

if((fp=fopen(“a2.txt”,”r”))==NULL)

{ printf(“Can not open file!\n”); exit(1); }

else { fc(fp); fclose(fp); }

}

void fc(FILE \*fp1)

{ char c;

while((c=fgetc(fp1))!=’#’) putchar(c);

}

四、程序设计题

1. 编一个程序，从键盘输入200个字符，存入名为“f1.txt”的磁盘文件中。
2. 从上一题中建立的名为“f1.txt”的磁盘文件中读取120个字符，并显示在屏幕上。
3. 编一个程序，将磁盘中当前目录下名为“ccw1.txt”的文本文件复制在同一目录下，文件名改为“ccw2.txt”。
4. 编程序对名为“CCW.TXT”的磁盘文件中“@”之前的所有字符加密，加密的方法是每个字节的内容减10。
5. 编程序对上一题中名为“CCW.TXT” 的磁盘文件中“@”字符之前的所有字符解密并显示。
6. 输入100个学生的信息（含学号、姓名、年龄、7科成绩、总分），统计所有学生的总分，然后存入磁盘二进制数据文件student.dat中。然后再读取该文件，寻找总分最高的学生并输出该学生的所有信息。
7. 编一个程序，统计文本文件中的单词个数。被统计文件的文件名可由键盘任意输入。
8. 有两个磁盘文件“f1.txt”和“f2.txt”各放一行字母，今要求把这两个文件中的字母合并，按字母顺序排列后，输出到新文件“f3.txt”中去。

**【答案】**

1. 单项选择题

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |  |  |  |  |  |
| A | C | B | B | C |  |  |  |  |  |

二、填充题

1. fopen fclose

2. 文本 二进制 文件结束标志 非零

3. ch=fgetc(fp) fscanf(fp,”%c”,&ch) fputc(ch,fp) fprintf(fp,”%c”,ch)

4. fp n-1 str fputs 写入文件的实际字符数

5. 文件型指针变量 stdio.h

6. rewind()

7. (!feof(fp))

8. ((fp=fopen(“stu.dat”,”rb”))==NULL) fread(stud,sizeof(struct student),N,fp)

三、程序分析题

1．该程序的功能是：从键盘读入一个字符，如果是小写字母，则变成大写输出到屏幕上，否则原样输出。当打入ctrl-z（即EOF) 时，输入结束。

2．该程序的功能是：统计文本文件fname.txt中的空格字符数。

3．该程序的功能是：把C盘根目录下的tc目录中的文件p1.c复制到A盘的根目录下，取同样的文件名p1.c。如果C盘中的p1.c文件超过1000字节，则最多复制1000字节。

4．该程序先打开a1.txt文件，然后调用函数fc()，把文件中#号字符以前的内容在屏幕上现示。再打开a2.txt文件，然后调用函数fc()，把文件中#号字符以前的内容在屏幕上接着现示。因此程序运行后输出 121314252627

四、程序设计题

1. 解

#include “stdio.h”

main()

{ FILE \*fp;

int i;

if((fp=fopen(“f1.txt”,”w”))==NULL)

{ printf(“Can not open file!\n”); exit(1); }

for( i=1;i<=200;i++) fputc(getchar(),fp);

fclose(fp);

}

1. 解

#include “stdio.h”

main()

{ FILE \*fp;

int i;

if((fp=fopen(“f1.txt”,”r”))==NULL)

{ printf(“Can not open file!\n”); exit(1); }

for( i=1;i<=120;i++) putchar(fgetc(fp));

fclose(fp);

}

1. 解

#include “stdio.h”

main()

{ FILE \*f1, \*f2;

if((f1=fopen(“ccw1.txt”,”r”))==NULL)

{ printf(“Can not open file!\n”); exit(0);}

if((f2=fopen(“ccw2.txt”,”w”))==NULL)

{ printf(“Can not open file!\n”); exit(0);}

while(!feof(f1)) fputc(fgetc(f1),f2);

fclose(f1); fclose(f2);

}

1. 解

设计思路：从文件CCW.TXT中读一个字符（字节），减10后写到一临时文件TMPCCW.TXT，直到读到字符‘@’。然后两文件反绕到头，从文件TMPCCW.TXT中读一个字符，写回到文件CCW.TXT，把原字符复盖。因此两文件用r+和w+的形式打开。

#include <stdio.h>

main()

{ FILE \*fp1,\*fp2;

char ch;

if((fp1=fopen("ccw.txt","r+"))==NULL)

{ printf("Can not open file!\n"); exit(0);}

if((fp2=fopen("tmpccw.txt","w+"))==NULL)

{ printf("Can not open file!\n"); exit(0);}

while((ch=fgetc(fp1))!='@') { fputc(ch-10,fp2);}

rewind(fp1); rewind(fp2);

do { ch=fgetc(fp2); if(!feof(fp2)) fputc(ch,fp1);

}while(!feof(fp2)) ;

fclose(fp1); fclose(fp2);

}

1. 解

#include <stdio.h>

main()

{ FILE \*fp1;

char ch;

if((fp1=fopen("ccw.txt","r"))==NULL)

{ printf("Can not open file!\n"); exit(0);}

while((ch=fgetc(fp1))!='@') putchar(ch+10);

fclose(fp1);

}

6． 解

方法一、

定义一个结构变量，每输入一个学生信息，即存入磁盘二进制数据文件student.dat中。建立存有100个学生信息的文件后，设计一个总分为0的hs学生，再打开读取该文件，每读入一个学生信息，即与hs比较，寻找总分最高的学生在hs，并输出该学生的所有信息。

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

void main()

{ struct student {

int sno;

char sn[20];

int sage;

float sg[7];

float sum;

};

struct student s,t,hs;

FILE \*fp;

int i,j;

if((fp=fopen("student.dat","w"))==NULL)

{ printf("Can not open file!\n"); exit(0); }

printf("以 学号\n 姓名\n 年龄\n成绩1 成绩2 成绩3 成绩4 成绩5 成绩6

成绩7 的形式输入:\n");

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

{ printf("学生%d:\n",i+1);

scanf("%d",&s.sno);

scanf("%s",s.sn);

scanf("%d",&s.sage);

s.sum=0;

for(j=0;j<7;j++)

{ scanf("%f",&s.sg[j]);

s.sum=s.sum+s.sg[j];

}

fwrite(&s,sizeof(struct student),1,fp);

}

fclose(fp);

if((fp=fopen("student.dat","r"))==NULL)

{ printf("Can not open file!\n"); exit(0); }

hs.sum=0;

while(!feof(fp))

{ fread(&t,sizeof(struct student),1,fp);

if(t.sum>hs.sum) {

hs.sno=t.sno;

strcpy(hs.sn,t.sn);

hs.sage=t.sage;

for(j=0;j<7;j++) hs.sg[j]=t.sg[j];

hs.sum=t.sum;

}

}

printf("总分最高的学生是 ");

printf("学号: %d 姓名: %s 年龄: %d\n",hs.sno,hs.sn,hs.sage);

printf("成绩1 成绩2 成绩3 成绩4 成绩5 成绩6 成绩7 总分\n");

for(j=0;j<7;j++) printf("%5.1f ",hs.sg[j]);

printf("%5.1f\n",hs.sum);

}

方法二、

定义一个100个元素的结构数组s，输入100个学生的信息后，一次存入磁盘二进制数据文件student.dat中。再打开该文件，读入100个学生信息到100个元素的结构数组t中。数组t中，用结构指针sp和hsp寻找总分最高的学生，使hsp指向它，并输出该学生的所有信息。

void main()

{

struct student {

int sno;

char sn[20];

int sage;

float sg[7];

float sum;

};

struct student s[100],t[100],\*sp,\*hsp;

FILE \*fp;

int i,j;

float hsum=0;

if((fp=fopen("student.dat","w"))==NULL)

{ printf("Can not open file!\n"); exit(0); }

printf("以 学号\n 姓名\n 年龄\n成绩1 成绩2 成绩3 成绩4 成绩5 成绩6

成绩7 的形式输入:\n");

for(i=0,sp=s;i<100;i++,sp++)

{ printf("学生%d:\n",i+1);

scanf("%d",&sp->sno);

scanf("%s",sp->sn);

scanf("%d",&sp->sage);

sp->sum=0;

for(j=0;j<7;j++)

{ scanf("%f",&sp->sg[j]);

sp->sum=sp->sum+sp->sg[j];

}

}

fwrite(s,sizeof(struct student),100,fp);

fclose(fp);

if((fp=fopen("student.dat","r"))==NULL)

{ printf("Can not open file!\n"); exit(0); }

fread(t,sizeof(struct student),100,fp);

for(i=0,hsp=t,sp=t;i<100;i++,sp++)

{ if(sp->sum>hsum) {

hsum=sp->sum;

hsp=sp;

}

}

printf("总分最高的学生是 ");

printf("学号: %d 姓名: %s 年龄: %d\n",hsp->sno,hsp->sn,hsp->sage);

printf("成绩1 成绩2 成绩3 成绩4 成绩5 成绩6 成绩7 总分\n");

for(j=0;j<7;j++) printf("%5.1f ",hsp->sg[j]);

printf("%5.1f\n",hsp->sum);

fclose(fp);

}

7. 解

#include <stdio.h>

void main()

{ char fname[20];

FILE \*fp;

int num=0, word=0;

char ch;

printf("输入要统计的文件名: ");

scanf("%s",fname);

if((fp=fopen(fname,"r"))==NULL)

{ printf("打不开文件 %s \n",fname); exit(0); }

while(!feof(fp))

{ ch=fgetc(fp);

if(ch==' ') word=0;

else if(word==0)

{ word=1;

num++;

}

}

printf("%s 文件中有 %d 个单词\n",fname,num);

fclose(fp);

}

8． 解

#include <stdio.h>

main()

{ FILE \*fp1,\*fp2,\*fp3;

char a[160],ch;

int i=0,j,n;

if((fp1=fopen("f1.txt","r"))==NULL)

{ printf("打不开文件f1.txt!\n"); exit(0); }

if((fp2=fopen("f2.txt","r"))==NULL)

{ printf("打不开文件f2.txt!\n"); exit(0); }

if((fp3=fopen("f3.txt","w"))==NULL)

{ printf("打不开文件f3.txt!\n"); exit(0); }

while(!feof(fp1))

{ ch=fgetc(fp1);

if(!feof(fp1)) a[i++]=ch; /\*读文件f1.txt放到数组a中\*/

}

while(!feof(fp2))

{ ch=fgetc(fp2);

if(!feof(fp2)) a[i++]=ch; /\*读文件f2.txt接着放到数组a中\*/

}

a[i]='\0'; n=i;

for(i=0;i<n-1;i++) /\*对数组a中的字母排序\*/

for(j=i+1;j<n;j++)

if(a[i]>a[j]) { ch=a[i]; a[i]=a[j]; a[j]=ch; }

i=0;

while(a[i]!='\0')

{ fputc(a[i],fp3); i++; } /\*把数组a逐个字母写入文件f3.txt \*/

fclose(fp1); fclose(fp2); fclose(fp3);

}