实验三

王梓帆　518021910109

1 实验目的

进一步理解、使用和掌握文件系统调用、文件的标准子例程，能利用和选择这些基本的文件操作完成复杂的文件处理工作。

2 实验设计

2.1

编写一个文件复制的C语言程序：

使用文件的系统调用`read(fd, buf, nbytes)`，`write(fd, buf, nbytes)`编写一个文件的复制程序

打开源文件`test.txt`，以及创建和打开目标文件`test_copy.txt`。

```
int fd_r, fd_w, n;
fd_r = open(in_file, O_RDONLY);
fd_w = creat(out_file, S_IWUSR|S_IRUSR);
fd_w = open(out_file, O_WRONLY);
```

定义缓冲区`buf`

```
char buf[BUFSIZE];        // #define BUFSIZE 1024
```

循环从`fd_r`中读取字符到`buf`中，在写入`fd_w`。

```
while ((n=read(fd_r, buf, BUFSIZE)) > 0)
{
  write(fd_w, buf, n);
}
```

关闭文件

```
close(fd_r);
close(fd_w);
```

使用库函数`fread(buf, size, nitems, fp)`和`fwrite(buf, size, nitems, fp)`实现复制

`FILE`结构文件的创建与打开

```
FILE *fp_r, *fp_w;
char buf[BUFSIZE];
int n, fd_w;
fp_r = fopen(in_file, "r");
fd_w = creat(out_file, S_IWUSR|S_IRUSR);
fp_w = fopen(out_file, "w");
```

循环读取和写入

```
// 从fp_r中循环读取BUFSIZE个char，并写入fp_w中
while ((n=fread(buf, sizeof(char), BUFSIZE, fp_r)) > 0)
{
  fwrite(buf, sizeof(char), n, fp_w);
}
```

文件关闭

```
fclose(fp_r);
fclose(fp_w);
```

一次读写一个字节，比较两种方法的执行效率

`read`/`write`方法，`clock_t`和`clock()`定义在头文件`time.h`中。

```
clock_t start, end;
start = clock();
while ((n=read(fd_r, buf, 1)) > 0)
{
  write(fd_w, buf, n);
}
end = clock();           //记录主循环的起始时间
```

`fread`/`fwrite`方法

```
start = clock();
while ((n=fread(buf, sizeof(char), 1, fp_r)) > 0)
{
  fwrite(buf, sizeof(char), n, fp_w);
}
end = clock();
```

更改参数一次读写1024字节再次实验

2.2

使用`fscanf`和`fprintf`复制文件

```
while (fscanf(fp_r, "%s", buf) > 0)
{
  fprintf(fp_w, "%s\n", buf);
}
```

使用`fgetc`和`fputc`复制文件

```
while ((buf=fgetc(fp_r))!=EOF)
{
  fputc(buf, fp_w);
}
```

使用`fgets`和`fputs`复制文件

```
while ((retstring = fgets(buf, BUFSIZE-1, fp_r)))
{
  fputs(retstring, fp_w);
}
```

2.3 父子进程通过无名管道通信

创建管道

```
int chan[2];
pipe(chan);
```

父进程

关闭读管道，从源文件中读取到`buf`中，再从`buf`中读取到`chan[1]`中

```
if (fork())
{
  close(chan[0]);
  fp_r = fopen(in_file, "r");
  int n;
  char buf[BUFSIZE];
  while ((n=fread(buf, sizeof(char), BUFSIZE, fp_r))) {
    write(chan[1], buf, n);
  }
  close(chan[1]);
  fclose(fp_r);
}
```

子进程

关闭写管道，从`chan[0]`中读取到`buf`中，再从`buf`读取到目标文件，这里的`buf`和父进程的`buf`不同。

```
else {
  close(chan[1]);
  fd_w = creat(out_file, S_IWUSR|S_IRUSR);
  fp_w = fopen(out_file, "w");
  int n;
  char buf[BUFSIZE];
  while ((n=read(chan[0], buf, BUFSIZE)))
  {
    fwrite(buf, sizeof(char), n, fp_w);
  }
  close(chan[0]);
  fclose(fp_w);
}
```

两个独立程序使用有名管道通信

读源文件进程

```
int main(int argc, char const *argv[])
{
    int fd, n;
    char buf[BUFSIZE];
    FILE *fp_r;
    const char *in_file = argv[1];
    mknod("fifo", S_IFIFO|0666, 0);          //创建有名管道
    fp_r = fopen("test.txt", "r");
    fd = open("fifo", O_WRONLY);
    while ((n=fread(buf, sizeof(char), BUFSIZE, fp_r))) {        //将源文件的内容写到buf中
        write(fd, buf, n);                                       //将buf中写入管道线
    }
    close(fd);
    fclose(fp_r);
}
```

写目标文件进程

```
int main(int argc, char const *argv[])
{
    int fd, fd_w, n;
    char buf[BUFSIZE];
    FILE *fp_w;
    const char *out_file = argv[1];
    fd_w = creat(out_file, S_IWUSR|S_IRUSR);
    fp_w = fopen(out_file, "w");
    fd = open("fifo", O_RDONLY);                    //打开管道文件
    while ((n=read(fd, buf, sizeof(buf)))) {        //读取管道文件
```

```
            fwrite(buf, sizeof(char), n, fp_w);          //将管道线中内容写至目标文件
        }
        close(fd);
        fclose(fp_w);
    }
```

3 实验运行结果

`test.txt`

test.txt

aaaaaaafnwsndlnwwwwwwwdlncslkdncsdaskmmmmds
dsdmkkkkkkkkke
saaaaaaaaaaaaaaasnmmmmmmmmmmmmmmmmm

3.1

`read()`+`write()`

```
wangzifan@wangzifandeMacBook-Pro exp3 % ./copyfile test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro exp3 % diff test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro exp3 %
```

`fread()`+`fwrite()`

```
wangzifan@wangzifandeMacBook-Pro exp3 % ./copyfile_stream test.txt test_copy.txt

wangzifan@wangzifandeMacBook-Pro exp3 % diff test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro exp3 %
```

一次读写一字节(以系统时钟滴答计算)

`read()`+`write()`

```
wangzifan@wangzifandeMacBook-Pro test_time % ./copyfile test.txt test_copy.txt
system call: 140556.000000
wangzifan@wangzifandeMacBook-Pro test_time %
```

`fread()`+`fwrite()`

```
wangzifan@wangzifandeMacBook-Pro test_time % ./copyfile_stream test.txt test_cop
y.txt
stream I/O call: 9918.000000
wangzifan@wangzifandeMacBook-Pro test_time %
```

一次读取1024字节

```
wangzifan@wangzifandeMacBook-Pro test_time % ./copyfile test.txt test_copy.txt
system call: 527.000000
wangzifan@wangzifandeMacBook-Pro test_time % ./copyfile_stream test.txt test_cop
y.txt
stream I/O call: 68.000000
wangzifan@wangzifandeMacBook-Pro test_time %
```

3.2

`fscanf()`+`fprintf()`

```
wangzifan@wangzifandeMacBook-Pro 2 % ls
test.txt        test1.c         test2.c         test3.c
test1           test2           test3
wangzifan@wangzifandeMacBook-Pro 2 % ./test1 test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 2 % diff test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 2 %
```

`fgetc()`+`fputc()`

```
wangzifan@wangzifandeMacBook-Pro 2 % rm test_copy.txt
wangzifan@wangzifandeMacBook-Pro 2 % ./test2 test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 2 % diff test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 2 %
```

`fgets()`+`fpust()`

```
wangzifan@wangzifandeMacBook-Pro 2 % rm test_copy.txt
wangzifan@wangzifandeMacBook-Pro 2 % ./test3 test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 2 % diff test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 2 %
```

3.3

```
wangzifan@wangzifandeMacBook-Pro 3 % rm test_copy.txt
wangzifan@wangzifandeMacBook-Pro 3 % ls
test.txt        test1           test1.c
wangzifan@wangzifandeMacBook-Pro 3 % ./test1 test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 3 % diff test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 3 %
```

3.4

```
wangzifan@wangzifandeMacBook-Pro 4 % rm test_copy.txt
wangzifan@wangzifandeMacBook-Pro 4 % ./test1 test.txt
wangzifan@wangzifandeMacBook-Pro 4 %
```

执行完`./test1 test.txt`，写管道等待读管道，开启新的终端：

```
wangzifan@wangzifandeMacBook-Pro 4 % ./test2 test_copy.txt
wangzifan@wangzifandeMacBook-Pro 4 %
```

两边同时执行。

```
wangzifan@wangzifandeMacBook-Pro 4 % diff test.txt test_copy.txt
wangzifan@wangzifandeMacBook-Pro 4 %
```