## 实验十 管道

课程名称 操作系统原理实验

实验项目名称 实验十 管道

学生姓名 司晨旭 专业班级 计算2002 学号 32001019

1. 使用pipe创建管道，在父进程和子进程之间进行通信

|  |
| --- |
| 课本代码省略 |

|  |
| --- |
|  |

|  |
| --- |
| data\_processed = read(file\_pipes[0], buffer, BUFSIZ);  子进程读数据  data\_processed = write(file\_pipes[1], buffer,strlen(buffer));  父进程写数据 |

1. 用sort命令打开一个管道，然后对一个字符数组排序。

使用popen()创建的管道必须使用pclose()关闭。Pclose函数等待popen进程启动的进程运行结束才关闭文件流

|  |
| --- |
| 课本代码省略 |

|  |
| --- |
|  |

|  |
| --- |
| 可以看到运行结果为 进行了排序  ( pipe\_fp = popen("sort", "w")  打开管道写数据 |

1. 两个子进程之间通过管道传送数据

|  |
| --- |
| 课本代码省略 |

|  |
| --- |
|  |

|  |
| --- |
| 可以看到实现了pstree|grep login  dup(fd[1]);  管道的写入端改为文件描述符1  dup(fd[0]);  管道的读取端改为文件描述符0 |

1. 多进程的管道通信

|  |
| --- |
| 课本代码省略 |

|  |
| --- |
|  |

|  |
| --- |
| write(fd[1],outpipe,50);     */\*向管道写长为50字节的串\*/*  read(fd[0],inpipe,50);*/\*从管道中读长为50字节的串\*/* |

1. 编程题

|  |
| --- |
|  |
| 代码如下  #include <stdio.h>  #include <stdlib.h>  #include <signal.h>  #include <unistd.h>  int pid1, pid2;  int i;  char ch;  void scanfALine(char *result*[]);  int main(void)  {      int fd0[2], fd1[2];      char readpipe[200], movepipe[200];      pipe(fd0);      pipe(fd1);      if (pid1 = fork() > 0)      { *// yeye*          wait(0);          wait(0);          read(fd1[0], movepipe, 50);          printf("Print: %s\n", movepipe);          exit(0);      }      else      {          if (pid2 = fork() > 0)          { *// erzi*              wait(0);              read(fd0[0], movepipe, 50);              printf("Move: %s\n", movepipe);              scanfALine(movepipe);              write(fd1[1], movepipe, 50);              exit(0);          }          else          {  *// sunzi*              scanfALine(readpipe);              write(fd0[1], readpipe, 50);              exit(0);          }      }  }  *//输入一行的代码为网上的*  void scanfALine(char *result*[])  {      while (1)      {          scanf("%c", &ch);          if (ch == '\n')              break;  *result*[i++] = ch;      }  *result*[i] = '\0';  }  不明白  子进程中为何将内容覆盖掉了 |