

Operating System

Lab03: IPC and map reduce

|  |  |
| --- | --- |
| 学 院： | 计算机与信息技术学院 |
| 学生姓名： | 刘嘉鹏 |
| 学 号： | 20281319 |

**北京交通大学**

2022年11月

目录

[1. 管道-第一个子项目 3](#_Toc118526614)

[2. 问题分析和解决 3](#_Toc118526615)

[3. 运行结果和分析 3](#_Toc118526616)

[4. 总结与建议 5](#_Toc118526617)

# 管道-第一个子项目

管道，是一种半双工通信方式，也就是说，通信的两方一个只能读，一个只能写，这是无名管道，所以使用方式只能是在父子进程之间。

这里仅使用无名管道实现父子进程通信，管道的一般使用方式是进程在使用fork函数创建子进程先创建一个管道，该管道用于在父子进程之间通信，然后创建子进程，之后父进程，关闭管道的读端，子进程关闭管道的写端，或者反其道行之。（fd[0]是读端 fd[1]是写端）

如果某进程要读取管道中的信息那么应该先关闭fd[1]，如果要写管道数据则关闭fd[0]。

在这个项目中，父进程需要从文件中读取数据传输到子进程，子进程进行数据处理（分析）后再把信息传回父进程，因此可以设置两个管道，实现双向的通信。

# 问题分析和解决

1. 利用fets函数读取数据时，我发现如果数据最后一行是空回车，那么在最后一次调用fgets(buf,size,fp)时，不会对buf进行更新，这会使得把空回车符号当成上一行数据读入，是错误的。因此最后一行的回车处理 ：buffer[0] = '\0'。
2. 我创建了两个管道用以传输文件，但是二者不能同时运行，必须通过传输-分析-传输的方式。原因在于父进程读取源文件写入时，是持续过程，必须全部读取并写入完成（可能有等待子进程读管道的过程），才会执行后续从子进程获取结果的部分代码。若同时子进程也在分析并写入另一管道，但是父进程却没有进行读取该管道，那么可能会由于管道满而导致子进程阻塞。

# 运行结果和分析

编译并执行程序，需要添加两个命令参数，分别是文件路径和关键词。文本分析结果会输出到output.txt

对big.txt文件，采用project关键字执行情况如下：

|  |
| --- |
| 文本  描述已自动生成 |

对ANNA\_KARENINA.txt，anna作为关键字执行情况如下:

|  |
| --- |
| 可以看到程序运行无误，不过并没有清除行首空格来进行排序。 |

# 总结与建议

这次实验我全部完成的只有第一个子项目，套接字和共享内存部分、map-reduce都还没来得及完成，着实体会到了项目的难度。自学能力仍有待提高。

希望老师可以在此类较高难度的项目下多给一些参考资料。