实验四 进程通信

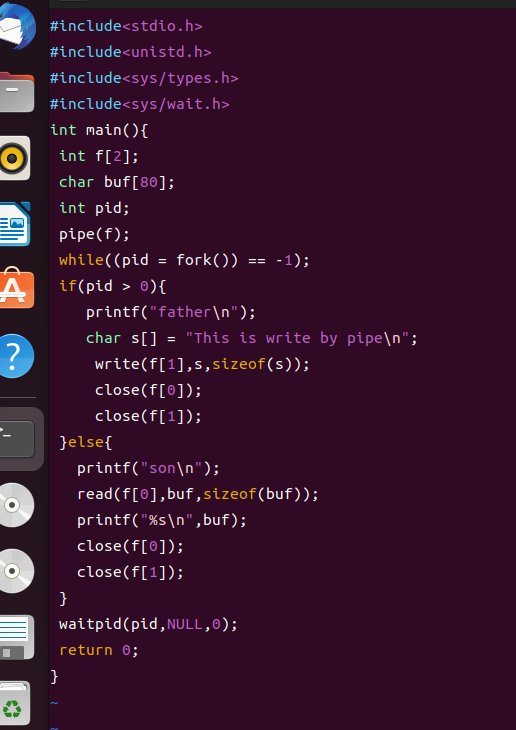
**学号： 20110031020 姓名： 许健 年级： 20计科**

**一．PIPE系统调用的基本使用方法**

**1.具体要求与步骤**

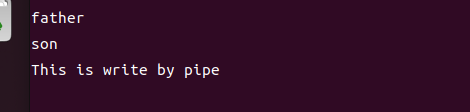
1. 编写一C/C++语言程序（程序名为basicpipe.c/basicpipe.cpp），使用系统调用pipe()实现基本的消息传递。
2. 多次连续反复运行这个程序，观察屏幕显示结果，试简单分析其原因。
3. 可以使用实验报告模板中所推荐的代码实现，但是要求为代码添加注释，对代码关键逻辑步骤进行解释。在代码头部添加如代码 1所示式样的头部版权声明。使用星号、井号、等号、破折号等各类符号对版权声明添加边框，并拼出19os的式样。

**2.实验结果截图**

1)代码：  


创建子进程，在子进程中将内容输入到管道中，然后再把管道中的内容在父进程中输出。

2）结果:



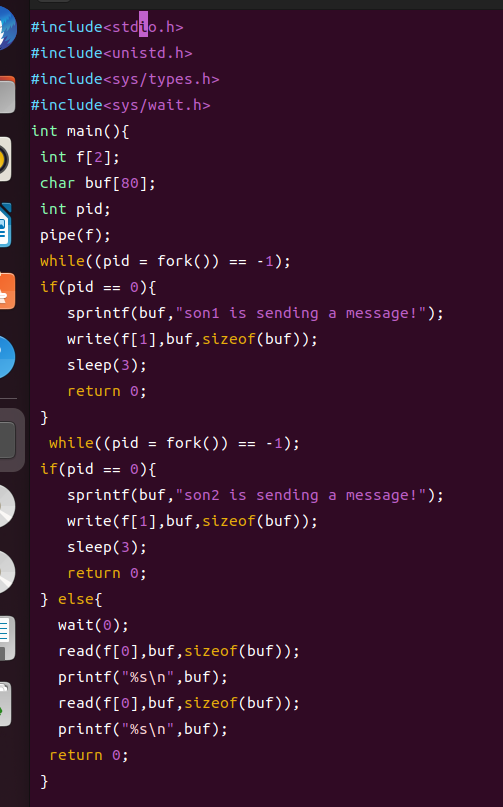
**二. PIPE实现进程间通信**

**1.具体要求与步骤**

1. 编写一C/C++语言程序（程序名为testpipe.c/testpipe.cpp），使用系统调用pipe( )尝试在多个进程间进行通信。
2. 要求实现的功能是，用管道来实现父子进程间通信。子进程向父进程发送字符串“[进程名称] is sending a message to parent!”；父进程则从管道中读出子进程发来的消息，并将其显示到屏幕上，然后终止。多次连续反复运行这个程序，观察屏幕显示结果的顺序，试简单分析其原因。
3. 可以使用实验报告模板中所推荐的代码实现，但是要求为代码添加注释，对代码关键逻辑步骤进行解释。在代码头部添加如代码 1所示式样的头部版权声明。使用星号、井号、等号、破折号等各类符号对版权声明添加边框，并拼出19os的式样。

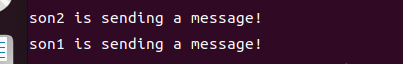
**2.实验结果截图**

1．1代码



创建两个子进程分别向管道输入内容，然后再在父进程中输出管道里的内容。

1.2结果

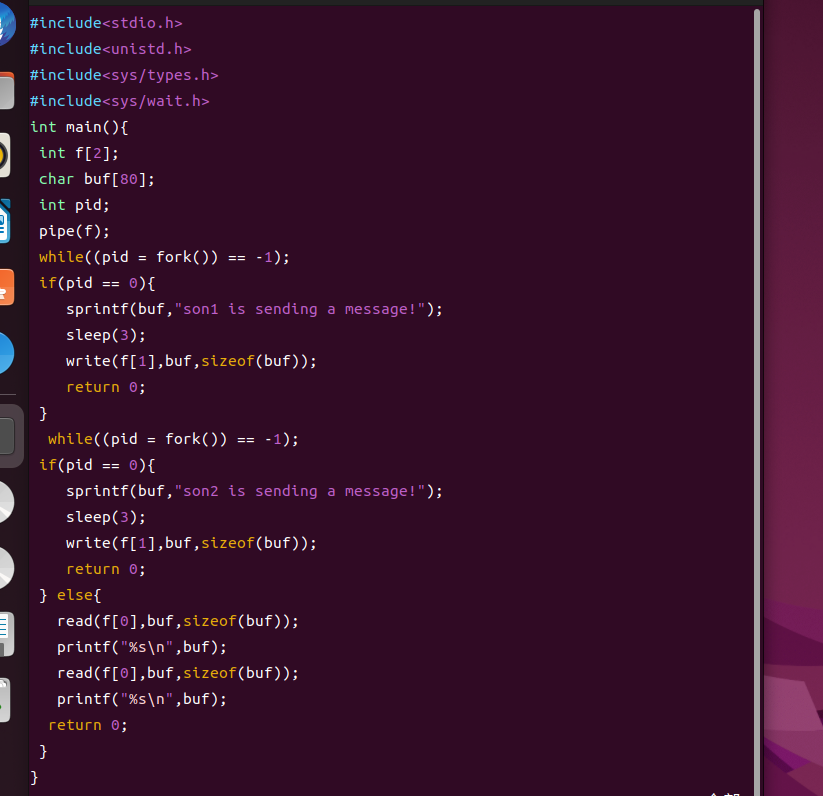




顺序不同是因为两个子进程的管道到达的顺序不同

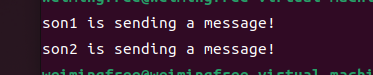
2.1

代码:



对源代码稍作修改，将父进程读取管道内容的时间早于子进程将内容写入管道的时间

2.2结果:



会停顿较长的时间，仍然会打印出结果，应是父进程在读取管道内容时，若此时管道没有内容，会阻塞自己