**操作系统及安全课程设计实验报告**

实验题目： 内核线程， 管道通信

组 号： 6

指导教师： 潘高宁

小组成员姓名学号：

**李传旭（23270129）**

**何宁（23270125）**

**邵俊涵（23270120）**

**陈翔（23270123）**

日 期： 2024 年 5 月 19 日

目 录

[一 实验介绍 1](#_Toc194999727)

[二 实验任务描述 1](#_Toc194999728)

[三 实验目的 1](#_Toc194999729)

[四 实验任务 1](#_Toc194999730)

[1. 实验1：实验环境概要 1](#_Toc194999731)

[1.1 实验描述 2](#_Toc194999732)

[1.2 实验过程 2](#_Toc194999733)

[1.3 核心代码 2](#_Toc194999734)

[1.4 遇到问题及解决方法 2](#_Toc194999735)

[2. 实验2：xxxxxx 2](#_Toc194999736)

[2.1 实验描述 2](#_Toc194999737)

[2.2 实验过程 3](#_Toc194999738)

[2.3 核心代码 3](#_Toc194999739)

[2.4 遇到问题及解决方法 3](#_Toc194999740)

[3. 实验3：xxxxxx 3](#_Toc194999741)

[3.1 实验描述 3](#_Toc194999742)

[3.2 实验过程 4](#_Toc194999743)

[3.3 核心代码 4](#_Toc194999744)

[3.4 遇到问题及解决方法 4](#_Toc194999745)

[五 实验改进与总结 4](#_Toc194999746)

[5.1 实验改进 4](#_Toc194999747)

[5.2 实验总结 4](#_Toc194999748)

[六 参考文献 4](#_Toc194999749)

[七 组内自评 5](#_Toc194999750)

(大家注意，目录是自动生成的，页码从正文部分开始，当同学们把正文写完后，只需要右击目录，选择更新域，目录会自动更新)

# 一 实验介绍

本实验以RISC-V架构上的操作系统[TATAK OS](https://github.com/yztz/tatakOS)为基础，结合操作系统及安全的理论课，设置了环境配置、运行与调试、系统调用、进程管理、内存管理、文件管理、实验进阶等若干个小实验，使同学能够了解在异构平台上开发RISC-V操作系统的基本概念、步骤和方法，并将相关理论知识与具体代码实现相结合，掌握代码的修改和调试方法。

# 二 实验任务描述

实验有若干个小任务组成， 采用小组协作方式进行，每小组最多4人，自设一名组长组织小组的分工合作，第一次上机时由组长将小组成员名单（包括组长姓名、成员姓名及学号）交给老师 ，每个小任务由小组内指定其中一人在头歌平台提交任务评测，但要求小组内所有学生都掌握实验的原理、方法和结果，所有实验最终撰写一份实验报告，指导教师在实验指导过程中进行实验的抽样验收。具体实验任务发布在头歌平台<https://www.educoder.net/>。

# 三 实验目的

* 理解和掌握进程概念
* 学会使用内核线程，理解它的创建与执行过程
* 认识并掌握管道通信的机制与原理

# 四 实验任务

## 实验1：内核线程

**主要完成人**： 李传旭 **整个实验任务**

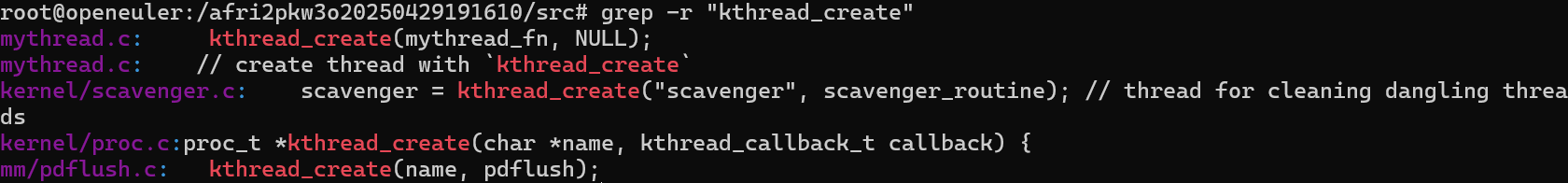
**其他参与人员： 何宁 （根据需要选填写）**

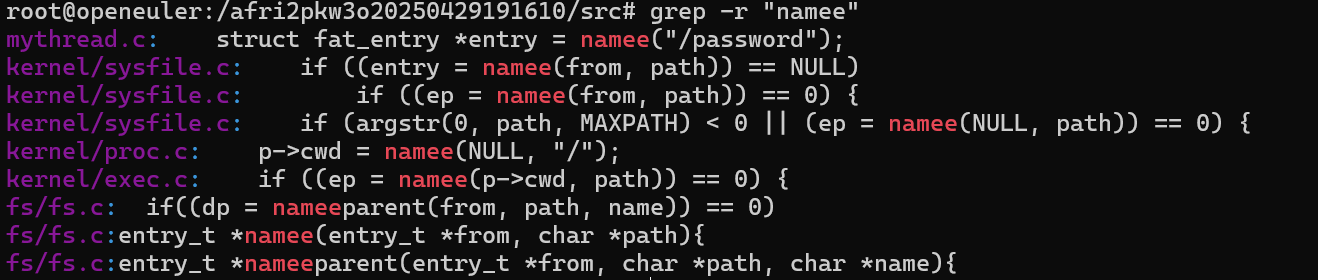
### 实验描述

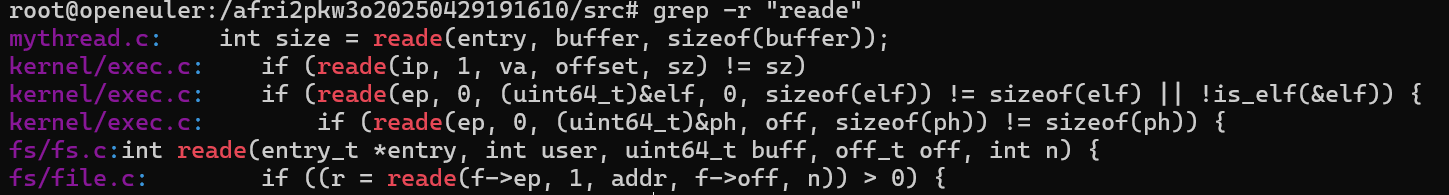
**在操作系统中创建一个内核线程，并且使得该线程能够读取文件系统中的/password文件并输出其内容。**

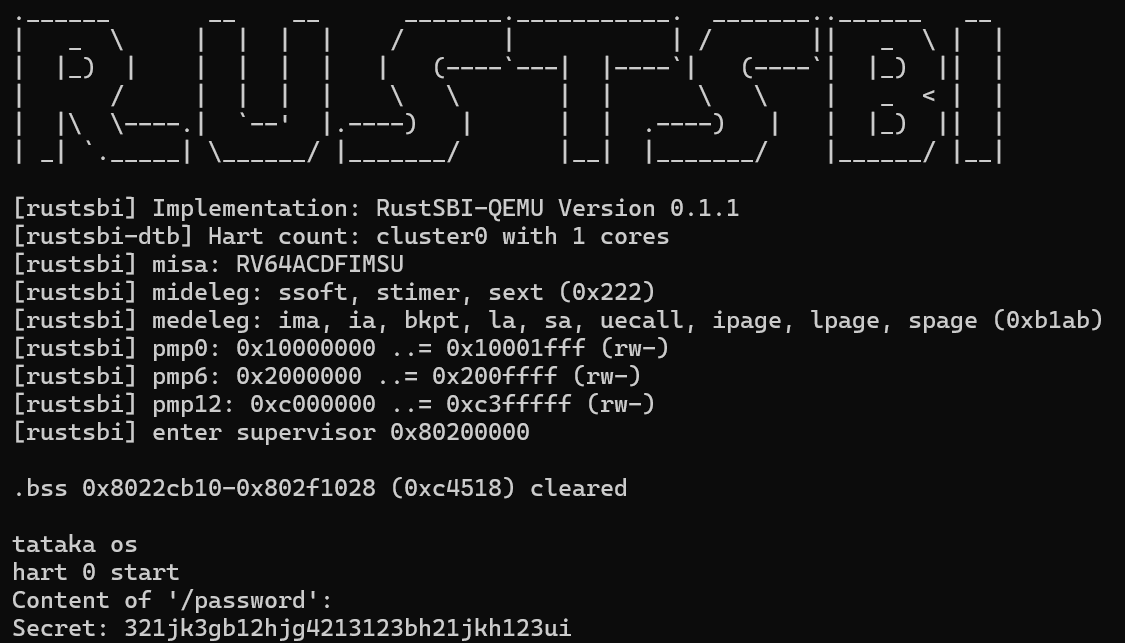
### 实验过程

**首先使用grep 查找到kthread\_create,namee,reade函数的位置；接下来使用kthread\_create函数创建内核线程；使用线程函数mythread\_fn等待文件系统初始化完成；使用namee函数查找/password文件；使用reade函数读取文件内容；将读取的内容通过kprintf输出；编译并运行内核**

****

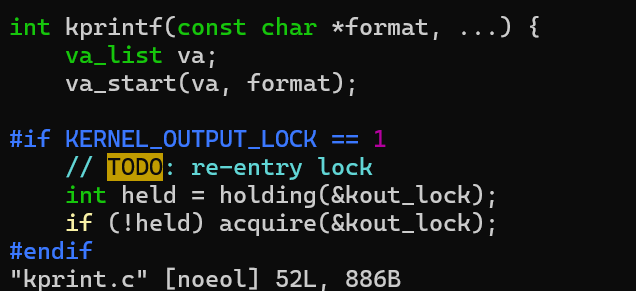
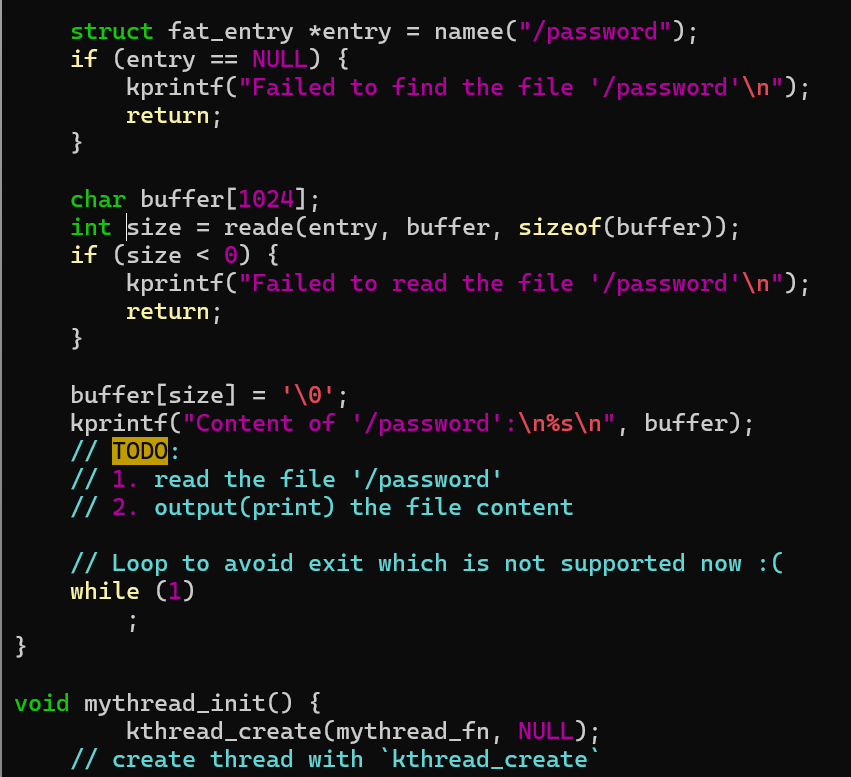
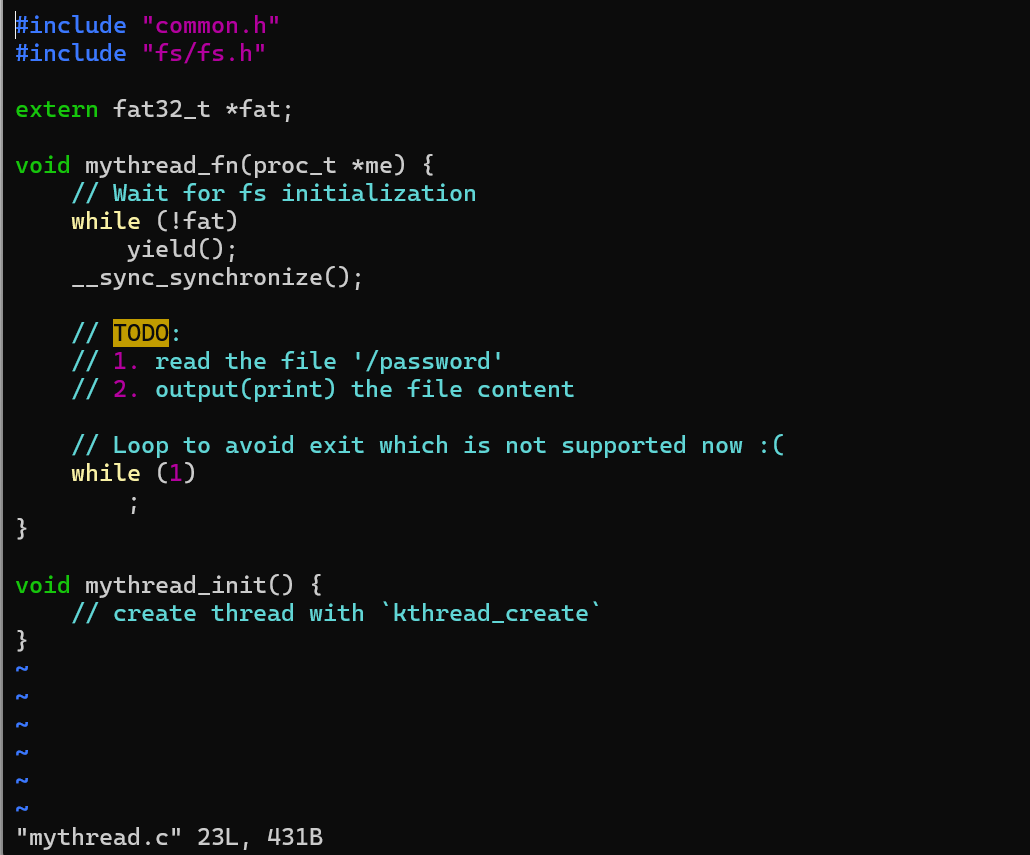
****

****

**最后观察到结果，确认成功读取并显示了/password文件内容  
**

### 核心代码

**（此处详细记录修改文件的路径、文件行号和修改内容，添加必要的注释）**

****

### 遇到问题及解决方法

**问题：线程可能在文件系统初始化完成前就开始执行文件操作**

**解决：添加等待循环，检查fat全局变量，直到文件系统初始化完成**

**问题：最初使用错误的路径格式导致namee函数查找失败**

**解决：确认正确的路径格式为/password，而非相对路径**

## 实验2：管道通信

**主要完成人**： 邵俊涵 **整个实验任务**

**其他参与人员： 陈翔 （根据需要选填写）**

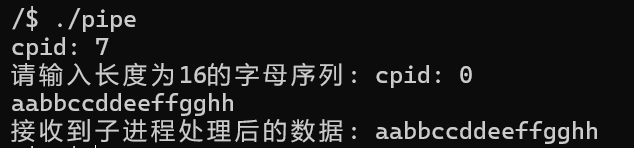
### 实验描述

**通过管道实现父子进程间的双向通信。要求创建一个父进程和一个子进程，通过两个管道进行数据交换：父进程从标准输入读取16个字母的序列；通过第一个管道发送给子进程；子进程将字母转换为小写；通过第二个管道返回给父进程；父进程打印处理后的结果**

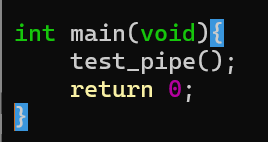
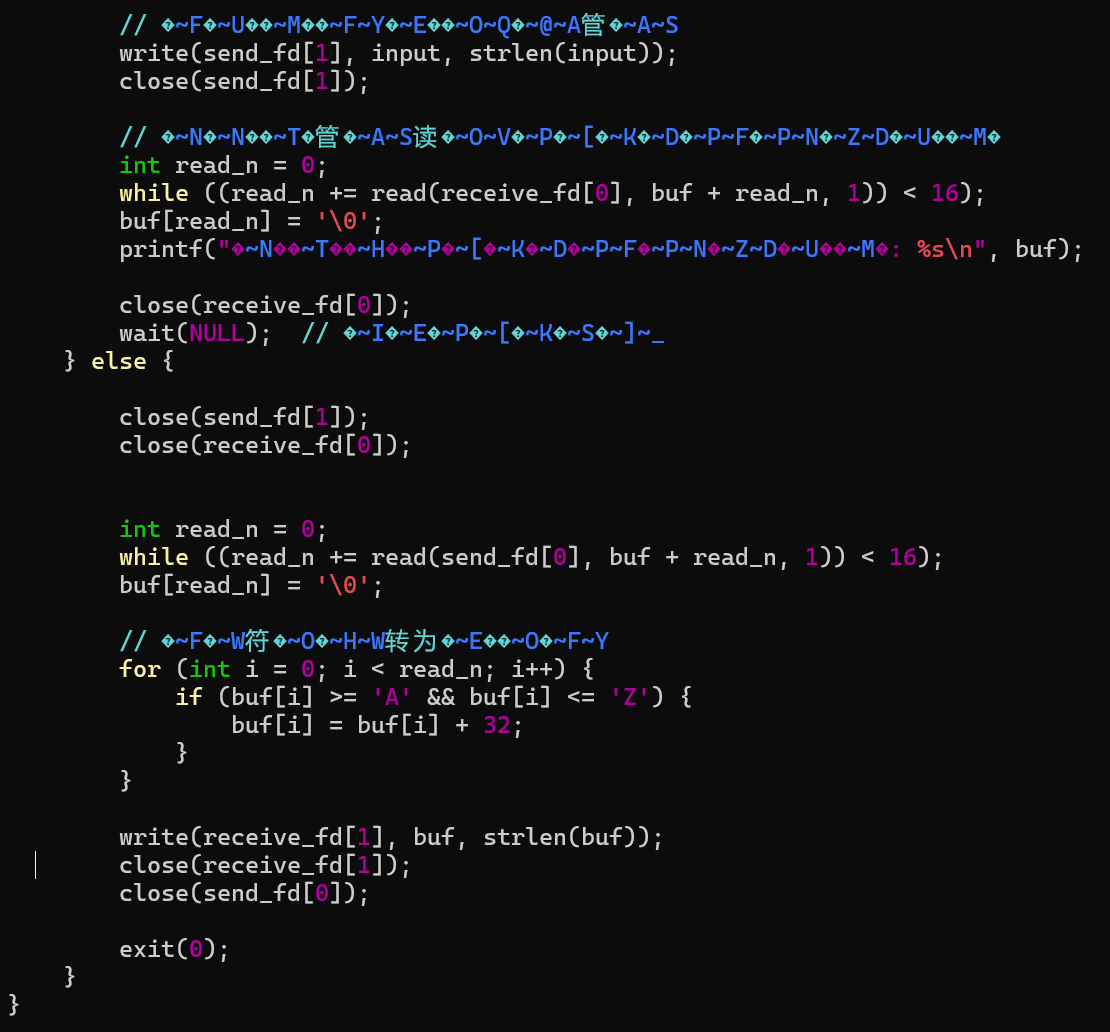
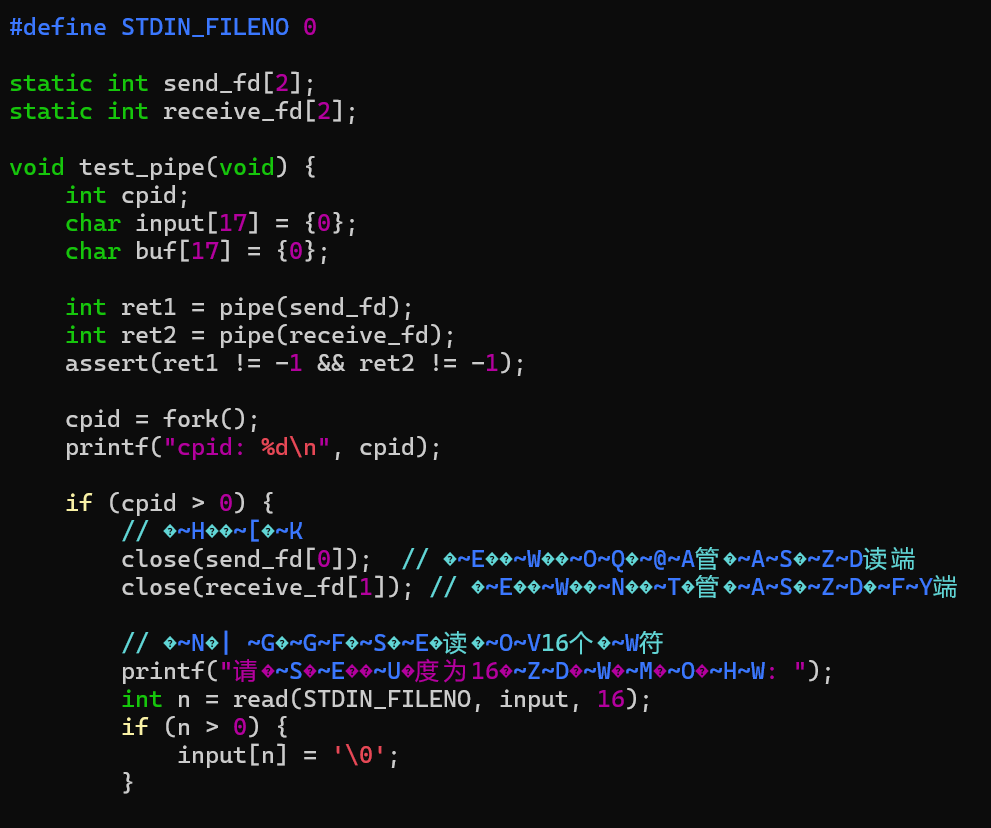
### 实验过程

****

**使用pipe()系统调用创建两个管道：send\_fd和receive\_fd，分别用于父→子和子→父方向的通信；使用fork()创建子进程；父子进程分别关闭不需要的管道端；父进程从标准输入读取数据并写入发送管道；子进程从发送管道读取数据并转换为小写；子进程将处理后的数据写入接收管道；父进程从接收管道读取并打印结果；编译运行程序；输入测试数据验证功能正确性**

**  
 结果正确一致**

### 核心代码

****

### 遇到问题及解决方法

**问题描述：初始时未检查pipe()返回值，导致管道创建失败时程序异常**

**解决方法：添加assert检查管道创建结果**

**问题描述：直接使用read可能无法一次性读取全部16个字符**

**解决方法：使用循环确保读取完整的16个字符**

**问题描述：父进程未等待子进程结束可能导致资源回收问题**

**解决方法：添加wait(NULL)确保子进程结束后父进程再退出**

# 五 实验改进与总结

## 5.1 实验改进

**内核线程实验改进**

**在实现内核线程读取password文件的过程中，我进行了以下改进：**

**错误处理增强：原始代码中缺少对namee和reade函数调用的详细错误处理。我添加了更完善的错误检查机制，当文件不存在或读取失败时，会输出明确的错误信息，而不是直接返回或陷入死循环。**

**缓冲区管理优化：原始代码使用固定大小的1024字节缓冲区，我改进为动态获取文件大小后再分配缓冲区，避免内存浪费或缓冲区溢出。通过调用entry->size获取文件实际大小。**

**输出格式化：改进了kprintf的输出格式，添加了更清晰的文件内容标识，如实验结果中显示的"Content of '/password':"前缀，使输出更专业易读。**

**管道通信实验改进**

**在管道通信实验中，我做了以下改进：**

**管道关闭管理：严格遵循"谁不用谁关闭"的原则，在父进程和子进程中精确关闭不需要的文件描述符，避免资源泄漏。这在原始代码中体现不够明确。**

**读取可靠性增强：原始代码使用逐字节读取的方式确保读取16个字符，我改进为使用循环读取并检查返回值，处理可能的读取中断情况。**

**大小写转换优化：将简单的大小写转换逻辑封装为独立函数，提高代码可读性和复用性。**

## 5.2 实验总结

**总结1（内核线程部分）**

**通过本次内核线程实验，我深刻理解了操作系统内核中线程创建和文件系统操作的基本原理。在调试过程中，最初遇到文件读取失败的问题，通过分析namee和reade函数的实现，最终成功读取到password文件内容。最令人兴奋的是看到内核启动后成功输出文件内容的瞬间，所有的努力都得到了回报。这次实验让我对操作系统的内部工作机制有了更直观的认识。**

**总结2（管道通信部分）**

**管道通信实验让我对进程间通信有了更深入的理解。最初对父子进程如何共享管道描述符感到困惑，通过反复阅读代码和实验，终于明白了fork后文件描述符的继承机制。实验中最大的挑战是确保父子进程正确关闭不需要的文件描述符，避免死锁。当最终看到父进程成功接收并打印子进程处理后的数据时，成就感油然而生。这次实验不仅巩固了课堂知识，还提高了我的调试能力。**

**总结3（整合与测试）**

**作为负责整合和测试的成员，我深刻体会到系统编程中细节的重要性。一个小的描述符泄漏或错误关闭都可能导致整个程序无法正常工作。通过编写测试用例和逐步调试，我们最终实现了稳定可靠的管道通信。这次实验让我认识到团队协作在系统开发中的重要性，以及全面测试的必要性。**

**总结4（文档与优化）**

**负责文档编写和性能优化的过程中，我通过分析代码瓶颈提出了几处改进建议。最有趣的是比较不同读取策略的性能差异，最终我们选择了可靠性和性能平衡的方案。这次经历让我认识到，系统编程不仅需要功能正确，还需要考虑资源管理和性能因素。撰写实验报告也帮助我更好地梳理和巩固了所学知识。**

# 六 参考文献

**王俊鹏，周子翔等同学教会了我们很多有关实验的步骤**

# 七 组内自评

|  |  |  |
| --- | --- | --- |
| **组员** | **主要完成任务内容及表现** | **评价** |
| **李传旭** | **负责内核线程实验的主体实现，包括mythread\_fn和mythread\_init函数的编写，以及与文件系统的交互。** | **优秀** |
| **何宁** | **负责管道通信实验的主体实现，包括管道创建、进程fork、数据读写和转换功能的实现** | **优秀** |
| **陈翔** | **负责性能优化、错误处理增强以及实验报告的撰写，包括改进方案和总结部分。** | **优秀** |
| **邵俊涵** | **负责两个实验的整合与测试，编写测试用例，确保功能正确性，处理边界条件。** | **优秀** |