



哈爾濱工業大學 (深圳)  
HARBIN INSTITUTE OF TECHNOLOGY

# 实验设计报告

开课学期: 大三秋季学期  
课程名称: 操作系统  
实验名称: 操作系统实验  
实验性质: 课内实验  
实验时间: 9月30日 地点: T2608  
学生班级: 2019级4班  
学生学号: 190110419  
学生姓名: 李怡凯  
评阅教师: \_\_\_\_\_  
报告成绩: \_\_\_\_\_

实验与创新实践教育中心印制

2021 年 10 月

## 一、 回答问题

### 1. 阅读 sleep.c, 回答下列问题

(1) 当用户在 xv6 的 shell 中, 输入了命令“sleep hello world\n”, 请问 argc 的值是多少, argv 数组大小是多少。

argc 值为 3, argv 数组大小为 3;

(2) 请描述 main 函数参数 argv 中的指针指向了哪些字符串, 他们的含义是什么。

指向在 shell 中敲入的参数, 以空格进行分割, 其中第 1 个参数为命令本身;

(3) 哪些代码调用了系统调用为程序 sleep 提供了服务?

exit(),atoi(),sleep()

### 2. 了解管道模型, 回答下列问题

(1) 简要说明你是怎么创建管道的, 又是怎么使用管道传输数据的。

使用 pipe()函数创建管道; 使用 write 函数向管道写数据, read 函数从管道读数据;

(2) fork 之后, 我们怎么用管道在父子进程传输数据?

在 fork 之前创建管道, 从而得到管道的文件描述符, 之后创建子进程, 即可利用该文件描述符在管道中传输数据。

(3) 试解释, 为什么要提前关闭管道中不使用的一端? (提示: 结合管道的阻塞机制)

当父进程没有关闭读写端的时候, 子进程读完管道里的内容后会阻塞等待父进程写端写入, 而程序的父进程如果没有写的操作, 那么子进程就会一直阻塞等待了。

## 二、 实验详细设计

### 1. pingpong

pingpong 程序通过管道实现父进程向子进程传送信息“ping”, 子进程接收到信息后打印自己的 pid 并输出接收到的信息, 然后再通过管道向父进程传送信息

“pong”，父进程接收到后同样打印自己的 pid 并输出接收到的信息。  
程序流程图如图 1 所示。

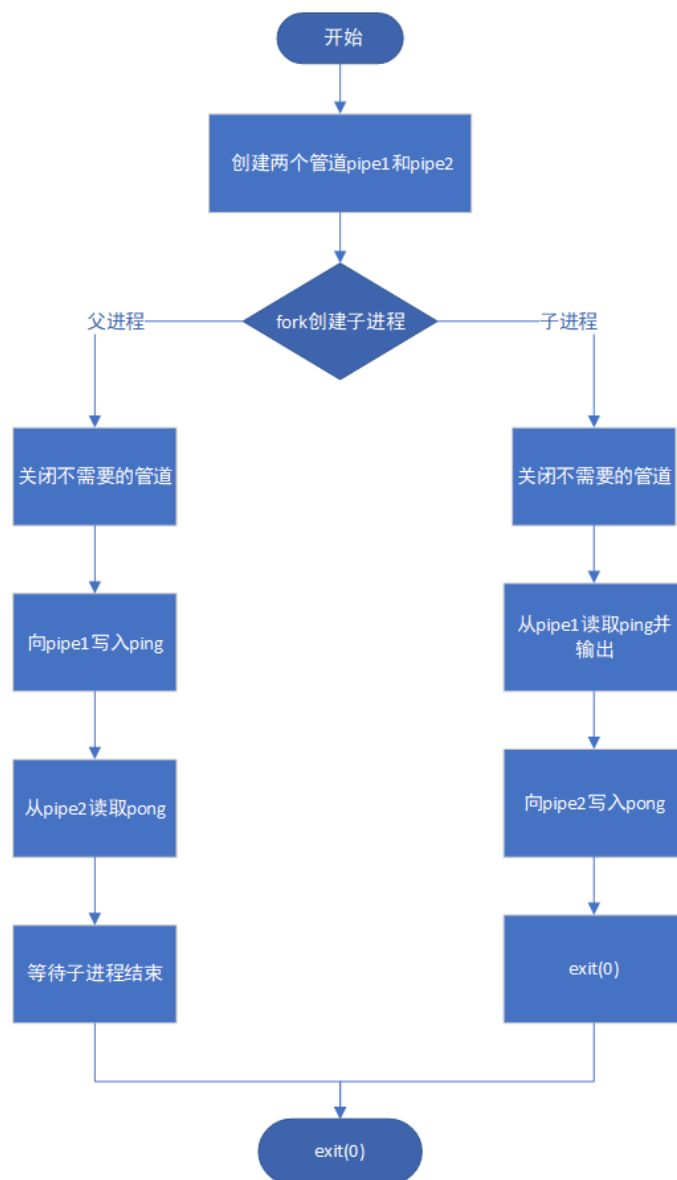


图 1 pingpong 程序流程图

## 2. 质数筛选

质数筛选程序通过多个进程与管道实现质数的筛选，每个子进程选出从管道读出的最小的数为质数，然后利用该质数筛选掉从管道读出的其他能够被该质数整除的数，剩余的数送入下一个管道由下一个进程处理，从而实现质数的筛选。

程序流程图如图 2，图 3 所示。其中第一个父进程将所有数字送入管道，同时最后送入一个数字 0，用于表示数字的结尾，子进程在读取到 0 之后就不会再继续建立管道与子进程。

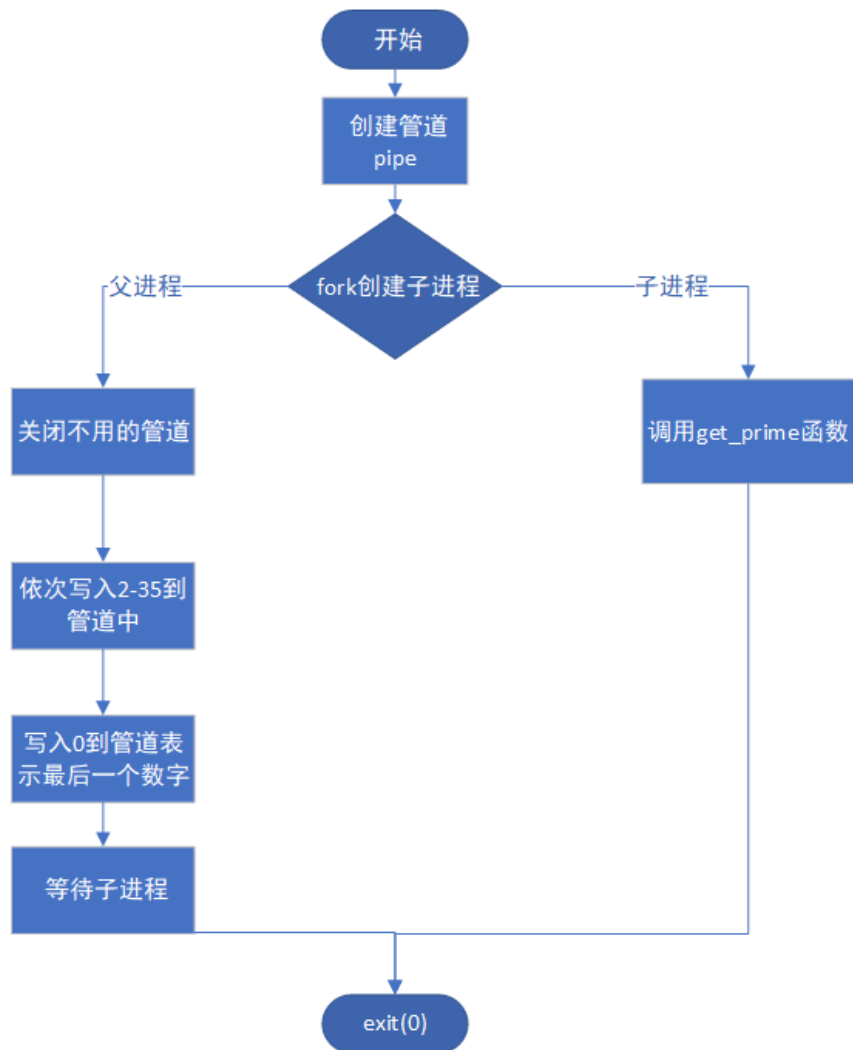


图 2 质数筛选主函数流程图

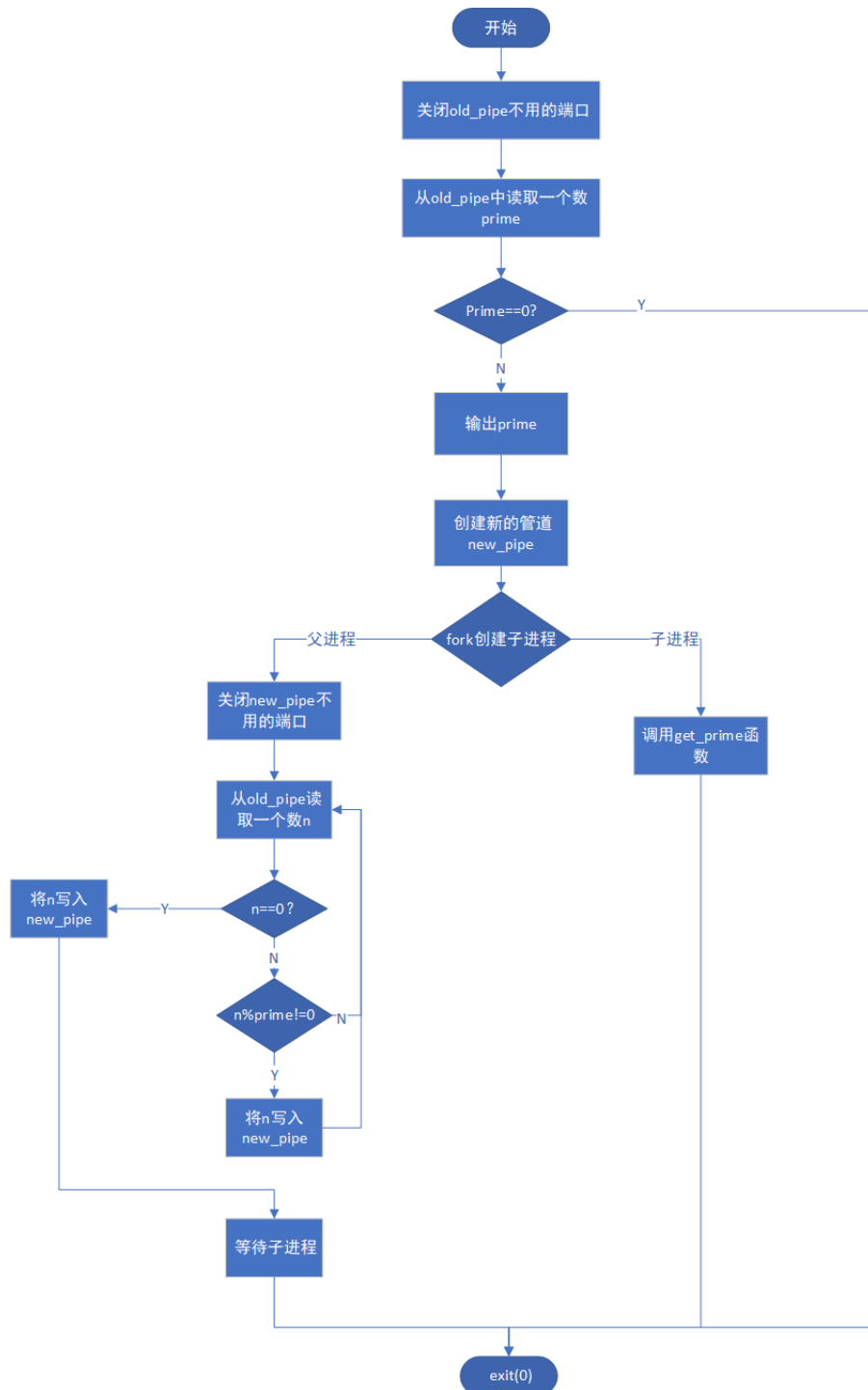


图3 质数筛选 get\_prime 函数流程图

### 3. find 实现

find 程序实现在给定的路径当中寻找给定的文件，如果路径下有文件夹则进入文件夹继续寻找，直到所有文件与文件夹都被检索过一次。

find 程序实现流程图如图4，图5所示。

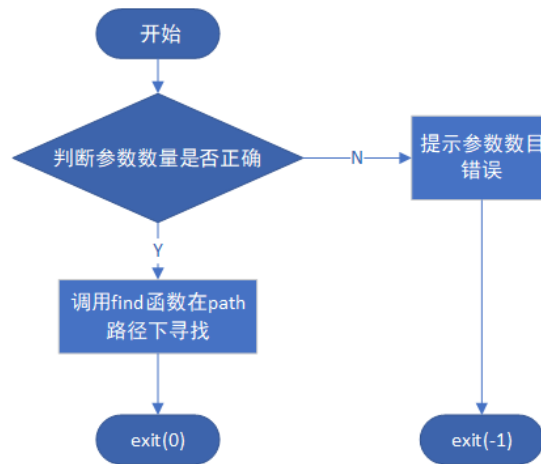


图 4 find 程序 main 函数流程图

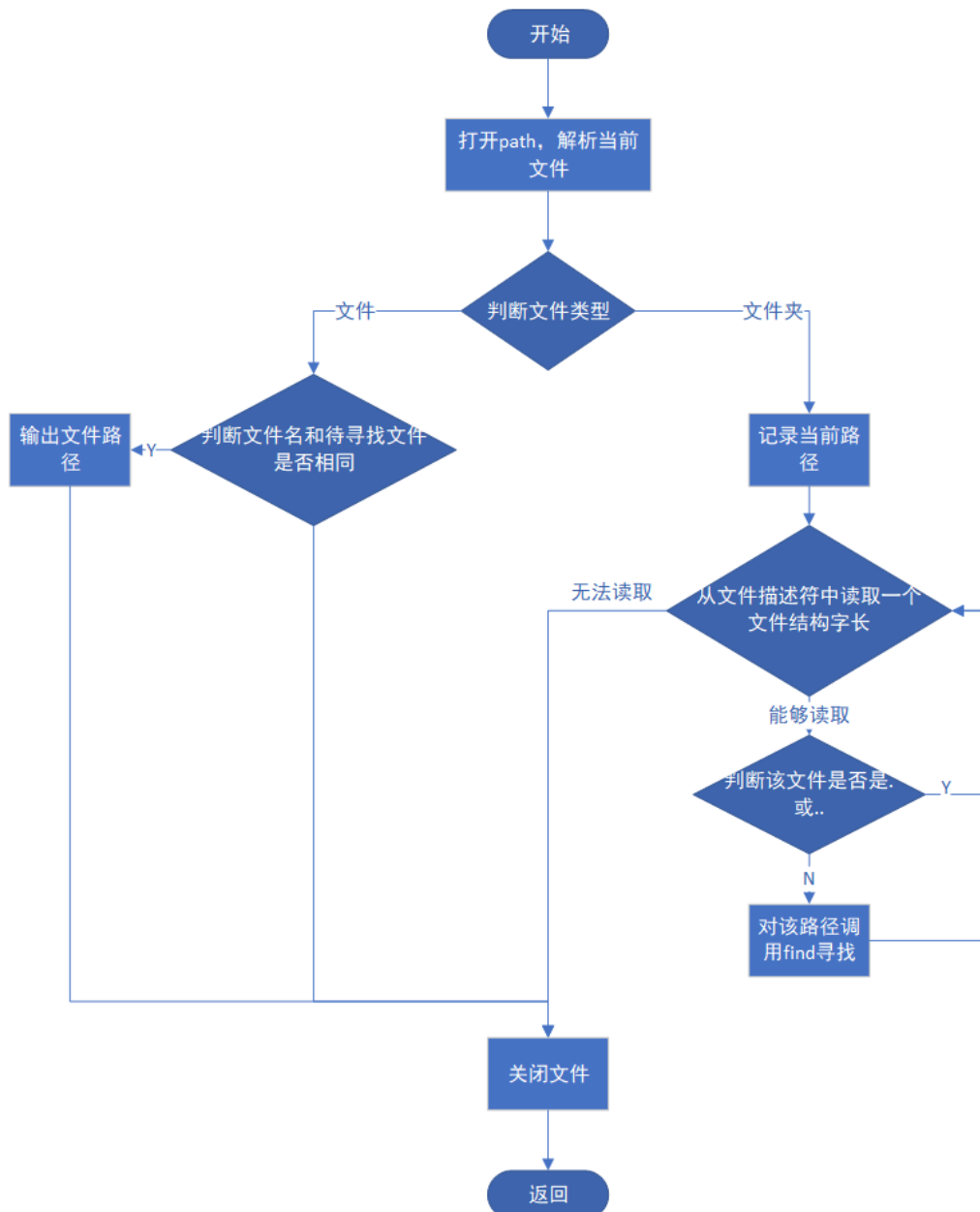


图 5 find 函数流程图

#### 4. xargs 实现

xargs 用于将标准输出作为输入送入下一个命令，用于补充部分无法通过管道进行参数传递的程序。

xargs 程序流程图如图 6 所示。

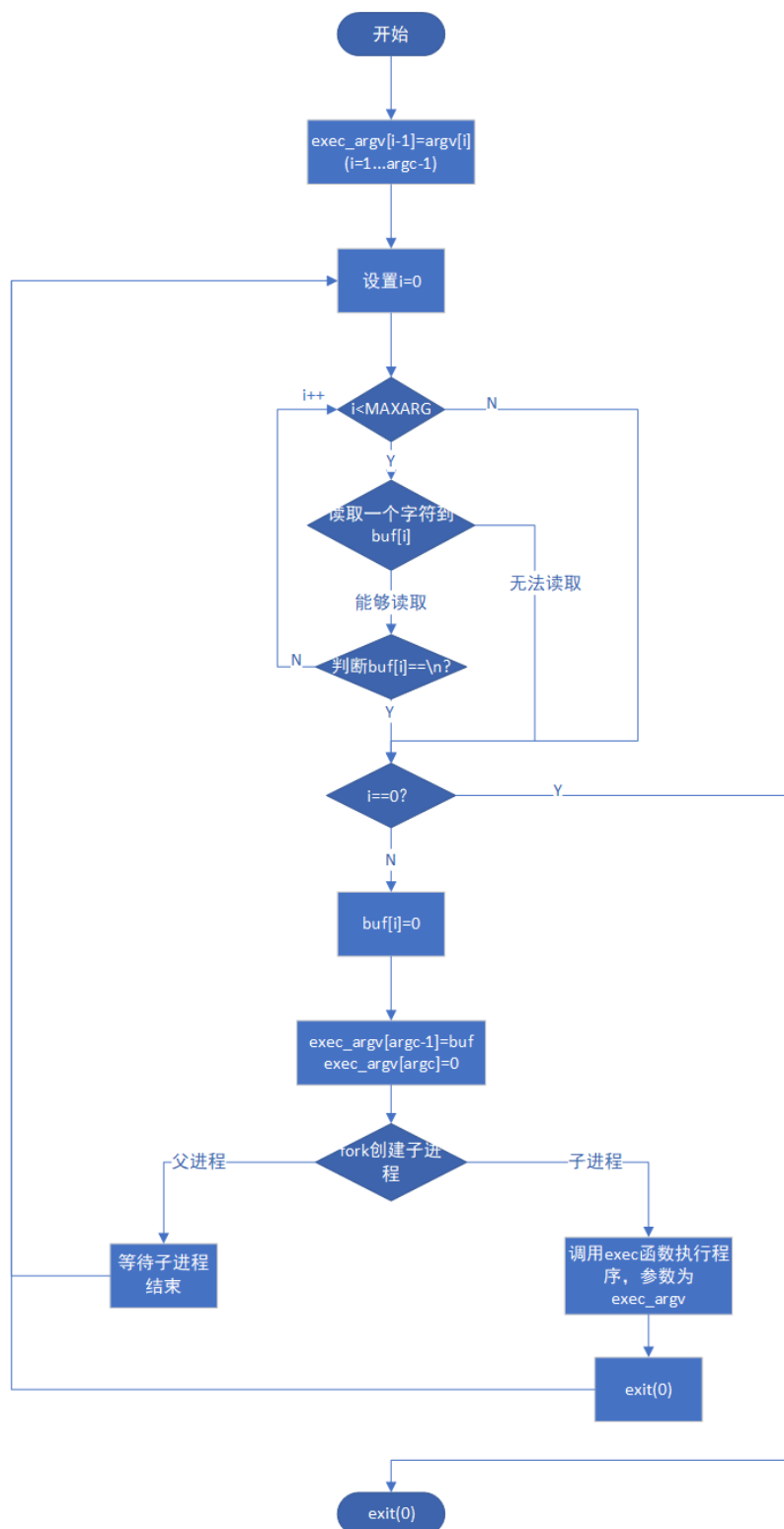


图 6 xargs 程序流程图

### 三、 实验结果截图

运行自动化测试脚本输出结果如图 7 所示。

```
190110419@OSLabExecNode0:~/xv6-labs-2020$ ./grade-lab-util
make: 'kernel/kernel' is up to date.
== Test sleep, no arguments == sleep, no arguments: OK (7.8s)
== Test sleep, returns == sleep, returns: OK (3.7s)
== Test sleep, makes syscall == sleep, makes syscall: OK (1.4s)
== Test pingpong == pingpong: OK (1.3s)
== Test primes == primes: OK (1.5s)
== Test find, in current directory == find, in current directory: OK (1.6s)
== Test find, recursive == find, recursive: OK (1.7s)
== Test xargs == xargs: OK (1.4s)
== Test time ==
time: OK
Score: 100/100
```

图 7 自动化测试脚本输出