**操作系统课程设计实验报告**

实验题目： Linux进程管理

组 号： 15

老师姓名： 袁文强

日 期： 2018 年 6月 8日

目 录

[一 实验小组分工 1](#_Toc516768816)

[二 题目介绍 1](#_Toc516768817)

[三 实验思路 1](#_Toc516768818)

[四 遇到问题及解决方法 1](#_Toc516768819)

[五 核心代码与实验结果及实验结果分析 1](#_Toc516768820)

[5.1 核心代码与实验结果 1](#_Toc516768821)

[5.2 实验结果分析 4](#_Toc516768822)

[六 实验改进与总结 4](#_Toc516768823)

[6.1 实验改进 4](#_Toc516768824)

[6.2 实验总结 4](#_Toc516768825)

[七 参考文献 4](#_Toc516768826)

# 一 实验小组分工

4-4 丁乃文

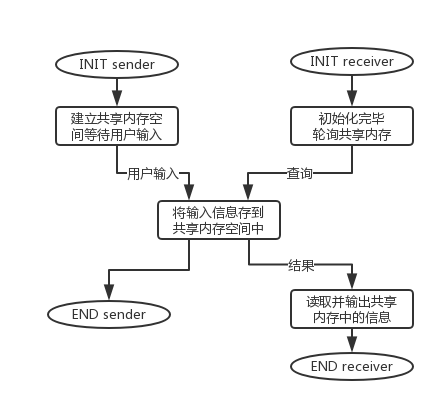
# 二 题目介绍

利用Linux的共享内存通信机制实现两个进程间的通信

# 三 实验思路

编写一个sender.c作为发送信息的一方，等待用户做一个输入，然后把用户输入的字符串保存到共享内存，当receive返回一个信息后根据信息返回输出。

编写一个receiver.c作为接收信息的一方，接收来自共享内存中的信息并将消息打印在终端上，然后删除共享内存并结束。



# 四 遇到问题及解决方法

之前做的时候没办法指定共享内存块，因此用init指定一下每次共享内存的地址就好了

# 五 核心代码与实验结果及实验结果分析

## 5.1 核心代码与实验结果

5.1.1创建信号量

void SemInit()*//创建信号量*

{

*/\**

*\* Funtion Prototype:*

*\**

*\*  sem\_t \*sem\_open(const char \*name, int oflag,*

*\* mode\_t mode, unsigned int value);*

*\**

*\* name     : MUTEX\_NAME    "mutex\_shm"*

*\* oflag    : O\_CREAT       Create and initialize it if not exist*

*\* mode\_t   : file perssion -rw-r--r--*

*\* value    : 1*

*\*/*

*if*((sem\_open(MUTEX\_NAME,O\_CREAT,0644,1)) < 0)

     {

        perror("sem\_open");

        exit(EXIT\_FAILURE);

     }

*if*((sem\_open(FULL\_NAME,O\_CREAT,0644,0)) < 0){

        perror("sem\_open");

        exit(EXIT\_FAILURE);

     }

}

5.1.2操作共享内存

int GetShmId(key\_t key)*//定义操作共享内存的函数*

{

    int shmid;

    shmid = shmget(key,SHM\_SIZE,IPC\_CREAT|0666);*//创建一个共享内存对象*

*if*(shmid < 0)*//如果不成功显示错误信息*

    {

        perror("Receiver: Shmget Error");

        exit(EXIT\_FAILURE);

    }

*return* shmid;

}

5.1.3封装PV操作

void P(sem\_t \*semPtr)

{

    sem\_wait(semPtr);                   *//int sem\_wait(sem\_t \*sem);*

}

void V(sem\_t \*semPtr)

{

    sem\_post(semPtr);                   *//int sem\_post(sem\_t \*sem);*

}

5.1.4读取共享内存

void ReadMessage()

{

    P(full);

    P(mutex);

    strcpy(result,shmptr);

    V(mutex);

}

5.1.5将内容放入共享内存

void SaveMessage()*//从读取的字符串中获取内容*

{

    P(mutex);

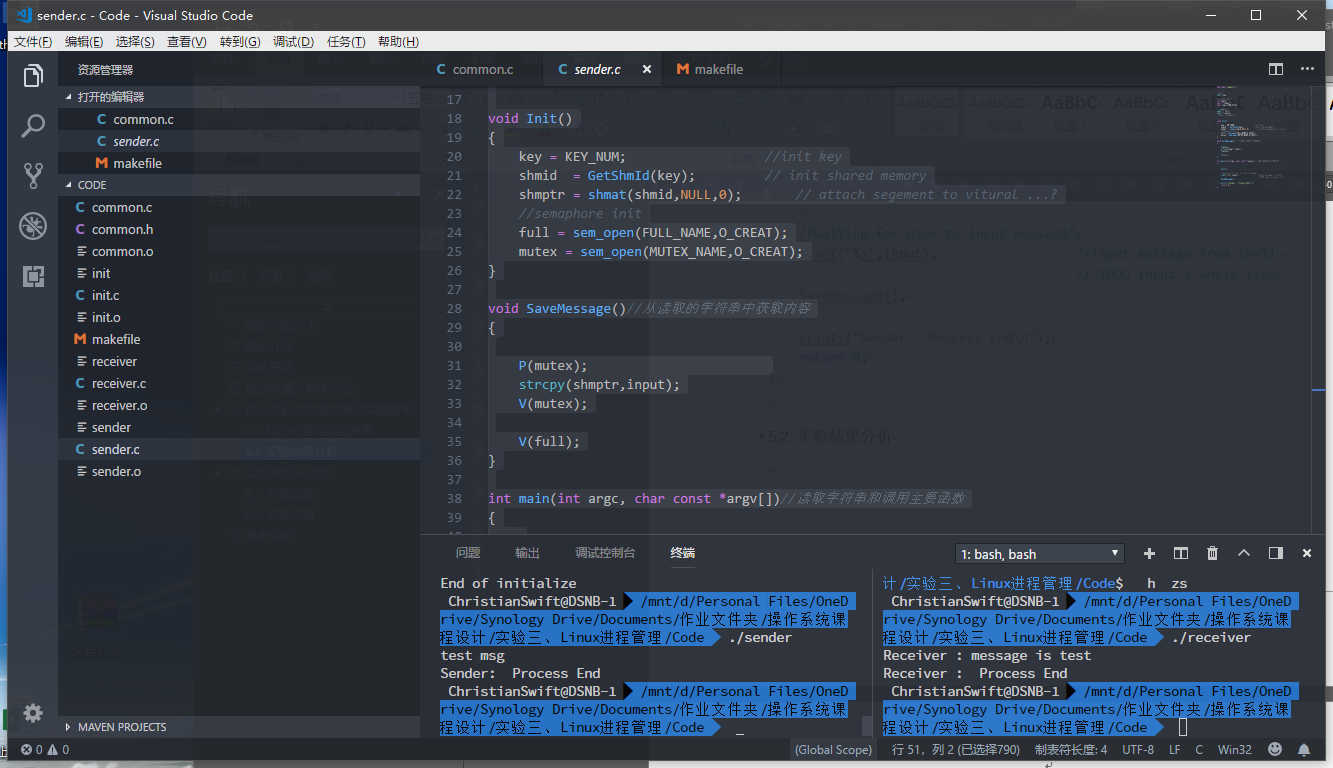
    strcpy(shmptr,input);

    V(mutex);

    V(full);

}

## 5.2 实验结果分析



# 六 实验改进与总结

## 6.1 实验改进

1、将公共函数写在了common.c中，并使用makefile管理编译

## 6.2 实验总结

通过这次上机实验，实现了通过Linux共享内存的通信功能，实现两个进程通信的另一种方式。实验中使用了信号量来保证两个进程的同步，让我对进程同步的概念进一步加深，更加了解了教材中关于信号量的说明。

# 七 参考文献