|  |
| --- |
| 北 京 邮 电 大 学  实 验 报 告  课程名称： 操作系统原理  院系：计算机学院（国家示范性软件工程学院）  班级： 2021211318  姓名： 李伟泽  学号： 2021211462  教师： 赵方  成绩：  2023年 春季学期 |
| 实验目的 （1）消息缓冲队列、共享存储区机制进行进程间的通信；  （2）理解通信机制。 实验环境 VMware Workstation Pro  Ubuntu22.10  Vi+GCC+GDB   1. **实验任务及内容**   1.使用消息缓冲队列来实现 client 进程和 server 进程之间的通信  server 进程先建立一个关键字为 SVKEY（如 75）的消息队列，然后等待接收类型为 REQ （例如 1）的消息；在收到请求消息后，它便显示字符串“serving for client”和接收到 的 client 进程的进程标识数，表示正在为 client 进程服务；然后再向 client 进程发送应答消息，该消息的类型是 client 进程的进程标识数，而正文则是 server 进程自己的标识 ID。client 进程则向消息队列发送类型为 REQ 的消息（消息的正文为自己的进程标识 ID） 以取得 sever 进程的服务， 并等待 server 进程发来的应答；然后显示字符串“receive reply from”和接收到的 server 进程的标识 ID。  *#include<sys/types.h>*  *#include<sys/msg.h>*  *#include<sys/ipc.h>*  *#include<stdio.h>*  *#include<unistd.h>*  *#include<stdlib.h>*  *#define SVKEY 75*  *struct mymsg{*  *long msg\_type;*  *char mytext[512];*  *int somethingelse;*  *float dollarval;*  *}msg;*  *int msgqid,pid,\*pint,i;*  *void client(){*  *msgqid=msgget(SVKEY,0666);*  *pid=getpid();*  *pint=(int\*)msg.mytext;*  *\*pint=pid;*  *msg.msg\_type=1;*  *msgsnd(msgqid,&msg,sizeof(int),0);*  *msgrcv(msgqid,&msg,250,pid,0);*  *printf("receive reply from pid=%d\n",\*pint);*  *exit(0);*  *}*  *void server(){*  *msgqid=msgget(SVKEY,0666|IPC\_CREAT);*  *msgrcv(msgqid,&msg,250,1,0);*  *pint=(int\*)msg.mytext;*  *pid=\*pint;*  *printf("serving for client pid=%d\n",pid);*  *msg.msg\_type=pid;*  *\*pint=getpid();*  *msgsnd(msgqid,&msg,sizeof(int),0);*  *exit(0);*  *}*  *int main(){*  *i=fork();*  *if(!i)*  *server();*  *i=fork();*  *if(!i)*  *client();*  *sleep(1);*  *}*  **运行结果：**   实验心得及体会 了解到使用msgget消息队列的创建，以及如何在多个进程间msgsnd发送消息和msgrcv接收消息。 |