Linux Programming LAB Week-8 19R21A0568 A. Narsimha Dhangar

# Write a C program for client server communication and print the message ? Program :

**server.c** #include<stdio.h> #include<netinet/in.h> #include<stdlib.h> #include<sys/socket.h> #include<sys/types.h>

int main(int argc,char const\* argv[]){

int servSockD = socket(AF\_INET,SOCK\_STREAM,0);

char serMsg[255]= "Message from server to the" " client\'Hello Client\' "; struct sockaddr\_in servAddr;

servAddr.sin\_family = AF\_INET; servAddr.sin\_port = htons(9001); servAddr.sin\_addr.s\_addr = INADDR\_ANY;

bind(servSockD, (struct sockaddr\*) & servAddr , sizeof(servAddr)); listen(servSockD,1);

int clientSocket = accept(servSockD,NULL,NULL); send(clientSocket,serMsg,sizeof(serMsg),0); return 0;

}

**client.c** #include<stdio.h> #include<netinet/in.h> #include<stdlib.h> #include<sys/socket.h> #include<sys/types.h>

int main(int argc,char const\* argv[]){

int sockD = socket(AF\_INET,SOCK\_STREAM,0); struct sockaddr\_in servAddr;

servAddr.sin\_family = AF\_INET; servAddr.sin\_port= htons(9001); servAddr.sin\_addr.s\_addr= INADDR\_ANY;

int connectStatus = connect(sockD,(struct sockaddr\*)&servAddr,sizeof(servAddr)); if(connectStatus == -1){

printf("Error...\n");

}

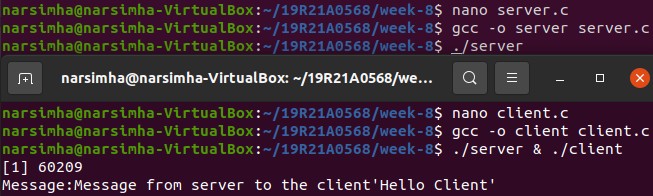
else{

char strData[255]; recv(sockD,strData,sizeof(strData),0); printf("Message:%s\n",strData);

}

return 0;

}



# Write a C program that illustrates semaphore ? Program:

#include<stdio.h> #include<unistd.h> #include<sys/types.h> #include<sys/ipc.h> #include<sys/sem.h> int main()

{

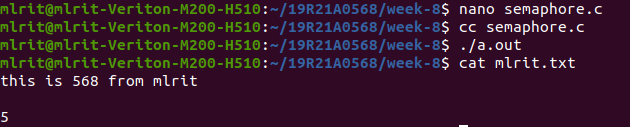
int semid; char a='5';

semid=semget((key\_t)3,1,IPC\_CREAT|0666); if(semid==-1)

printf("Error in creating semaphore\n"); semctl(semid,0,SETVAL,6);

struct sembuf sem\_op; FILE \*fd; fd=fopen("mlrit.txt","a"); sem\_op.sem\_num=0; sem\_op.sem\_op=-1; sem\_op.sem\_flg=0; semop(semid,&sem\_op,1); putc('\n',fd);

putc(a,fd); putc('\n',fd); fclose(fd);

}

# Write a C program that illustrates the message queues with a sender and receiver ?

**Program:**

# sender.c

#include <stdio.h> #include <sys/ipc.h> #include <sys/msg.h> #include <string.h> #include <stdlib.h> struct mesg\_buffer {

long mesg\_type; char mesg\_text[100];

} message; int main(){

key\_t key; int msgid;

key = ftok("progfile", 65);

msgid = msgget(key, 0666 | IPC\_CREAT); message.mesg\_type = 1;

printf("Write Data : "); gets(message.mesg\_text);

msgsnd(msgid, &message, sizeof(message), 0); printf("Data send is : %s \n", message.mesg\_text); return 0; }

# receiver.c

#include <stdio.h> #include <sys/ipc.h> #include <sys/msg.h> struct mesg\_buffer {

long mesg\_type; char mesg\_text[100];

} message; int main() {

key\_t key;

int msgid;

key = ftok("progfile", 65);

msgid = msgget(key, 0666 | IPC\_CREAT); msgrcv(msgid, &message, sizeof(message), 1, 0); printf("Data Received is : %s \n",message.mesg\_text); msgctl(msgid, IPC\_RMID, NULL);

return 0; }

