

OS lab submission-4

- Name : P K Navin Shrinivas
- Section : D
- SRN : PES2UG20CS237

Program 1 : Message Queues

sender code :

```
#include <stdio.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <sys/types.h>
#include <unistd.h>

struct my_msgbuf {
    long mtype;
    char mtext[200];
};

int main() {
    int msgid = msgget(ftok("haloo", 65), 0777);
    // ftok gets us a unique ID for our message queue
    // 0777 id the read write permissions for the queue
    struct my_msgbuf msg;
    printf("Enter text to append to queue : ");
    scanf("%[^\n]*c", msg.mtext);
    msg.mtype = 1; // must, can be 0 or greater
```

```

    msgsnd(msgid, &msg, sizeof(msg), 0);
    // msgsend ofc sends the message :)
}

```

rcv code :

```

#include <stdio.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <sys/types.h>
#include <unistd.h>
struct my_msgbuf {
    long mtype;
    char mtext[200];
};

int main() {
    struct my_msgbuf msg;
    int msgid = msgget(ftok("haloo", 65), 0777);
    char output[1000];
    msgrcv(msgid, &msg, sizeof(msg), 1, 0);
    // message rcv revices all the messages of type
    printf("%s", msg.mtext);
}

```

Screenshots :

```

/UE20CS254-05LAB/submission-4/messages_queues
→ messages_queues git:(main) X gcc reciever.c -o reciever
→ messages_queues git:(main) X gcc sender.c -o sender
→ messages_queues git:(main) X ./sender
Enter text to append to queue : hello world
→ messages_queues git:(main) X

/UE20CS254-05LAB/submission-4/messages_queues
→ messages_queues git:(main) X ./reciever
hello world
→ messages_queues git:(main) X

```

Program 2 : FIFO-named pipes

user1 code :

```
// This is a 2 pipe, only rcv from one side and send to other side  
// send side  
#include <fcntl.h>  
#include <stdbool.h>  
#include <stdio.h>  
#include <string.h>  
#include <sys/stat.h>  
#include <unistd.h>  
int main() {  
    int fd;  
    char *file = "/home/navin/hello";  
    mkfifo(file, 0777);  
    while (true) {  
        fd = open(file, O_WRONLY);  
        char input[80];  
        printf("Enter message : ");  
        fflush(stdout);  
        scanf("\t%[^\n]c", input);  
        write(fd, input, strlen(input) + 1);  
        close(fd);  
  
        fd = open("/home/navin/hello", O_RDONLY);  
        char output[80];  
        read(fd, output, sizeof(output));  
        printf("got back : %s \n", output);  
        close(fd);  
    }  
}
```

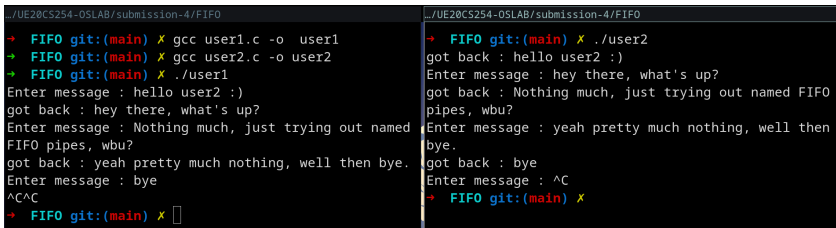
user2 code :

```
#include <sys/ipc.h>
#include <sys/msg.h>
#include <sys/types.h>
#include <unistd.h>

struct my_msgbuf {
    long mtype;
    char mtext[200];
};

int main() {
    struct my_msgbuf msg;
    int msgid = msgget(ftok("haloo", 65), 0777);
    char output[1000];
    msgrcv(msgid, &msg, sizeof(msg), 1, 0);
    // message rcv revices all the messages of type
    printf("%s", msg.mtext);
}
```

Screenshots :



```
.../UE20CS254-OSLAB/submission-4/FIFO
→ FIFO git:(main) X gcc user1.c -o user1
→ FIFO git:(main) X gcc user2.c -o user2
→ FIFO git:(main) X ./user1
Enter message : hello user2 :)
got back : hey there, what's up?
Enter message : Nothing much, just trying out named
FIFO pipes, wbu?
got back : yeah pretty much nothing, well then bye.
Enter message : bye
^C^C
→ FIFO git:(main) X []

.../UE20CS254-OSLAB/submission-4/FIFO
→ FIFO git:(main) X ./user2
got back : hello user2 :)
Enter message : hey there, what's up?
got back : Nothing much, just trying out named FIFO
pipes, wbu?
Enter message : yeah pretty much nothing, well then
bye.
got back : bye
Enter message : ^C
→ FIFO git:(main) X
```