

OS-Lab-7

Name: Anas Altaf

Roll.no: 22F-3639

Task-1

```
#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>
#include <string.h>

char messageP2toP1[] = "Message from P2!";
char messageP1toP2[] = "Message from P1!";

int main()
{
    char buf[1024];
    int pipeP2toP1[2];
    int pipeP1toP2[2];

    pipe(pipeP2toP1);
    pipe(pipeP1toP2);

    if (fork() != 0)
    {
        while (1)
        {
            read(pipeP2toP1[0], buf, 1024);
            printf("P1 received: %s\n", buf);

            write(pipeP1toP2[1], messageP1toP2, strlen(messageP1toP2) + 1);
        }
    }
    else
    {

```

```

while (1)
{

    write(pipeP2toP1[1], messageP2toP1, strlen(messageP2toP1) + 1);

    read(pipeP1toP2[0], buf, 1024);
    printf("P2 received: %s\n", buf);

}

}

return 0;
}

```

```

P1 received: Message from P2!
P1 received: Message from P2!
P1 received: Message from P2!
P1 received: Message from P2!
P1 received: Message from P2!
P1 received: Message from P2!
P1 received: Message from P2!
P1 received: Message from P2!
P1 received: Message from P2!

```

Task-2

Task-3

Producer:

```

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/ipc.h>

```

```
#include <sys/shm.h>
#include <sys/types.h>
#include <sys/sem.h>
#include <string.h>

#define SHM_SIZE sizeof(int)
#define SEM_KEY 1234

void P(int semid)
{
    struct sembuf p = {0, -1, 0};
    semop(semid, &p, 1);
}

void V(int semid)
{
    struct sembuf v = {0, 1, 0};
    semop(semid, &v, 1);
}

int main()
{
    key_t key = ftok("producer.c", 65);
    int shmid = shmget(key, SHM_SIZE, IPC_CREAT | 0666);
    int *shelf = (int *)shmat(shmid, NULL, 0);

    int semid = semget(SEM_KEY, 1, IPC_CREAT | 0666);
    semctl(semid, 0, SETVAL, 1);

    *shelf = 5;

    while (1)
    {
        P(semid);

        if (*shelf < 5)
        {
            (*shelf)++;
            printf("Producer: Added 1 item. Current shelf count: %d\n",
                *shelf);
        }
    }
}
```

```

    }

    V(semid);
    sleep(1);
}

shmdt(shelf);
shmctl(shmid, IPC_RMID, NULL);
return 0;
}

```

Consumer:

```

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <sys/types.h>
#include <sys/sem.h>
#include <string.h>

#define SHM_SIZE sizeof(int)
#define SEM_KEY 1234

void P(int semid)
{
    struct sembuf p = {0, -1, 0};
    semop(semid, &p, 1);
}

void V(int semid)
{
    struct sembuf v = {0, 1, 0};
    semop(semid, &v, 1);
}

```

```

int main()
{
    key_t key = ftok("producer.c", 65);
    int shmid = shmget(key, SHM_SIZE, 0666);
    int *shelf = (int *)shmat(shmid, NULL, 0);

    int semid = semget(SEM_KEY, 1, 0666);

    while (1)
    {
        P(semid);

        if (*shelf > 0)
        {
            (*shelf)--;
            printf("Consumer: Removed 1 item. Current shelf count: %d\n",
*shelf);
        }
        else
        {
            printf("Consumer: Blasted!! I knew I should have gone to
T---t.\n");
        }

        V(semid);
        sleep(2);
    }

    shmdt(shelf);
    return 0;
}

```

```
t1      t2.c  t3-producer  t4-reciever.c  test
xit@xit:/media/xit/3rd/University Tasks/FAST-BSE-5B/05 Lab/Lab_07$ ./t3-consumer
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4
Consumer: Removed 1 item. Current shelf count: 4

xit@xit:/media/xit/3rd/University Tasks/FAST-BSE-5B/05 Lab/Lab_07$ ./t3-producer
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
Producer: Added 1 item. Current shelf count: 5
```

Task-4

Bash : mkfifo fifo1

mkfifo fifo2

Sender:

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>

#define FIFO1 "fifo1"
#define FIFO2 "fifo2"
#define BUFFER_SIZE 1024

int main() {
    char message[BUFFER_SIZE];
    int fd;
```

```

while (1) {

    fd = open(FIFO1, O_WRONLY);

    printf("User 1: ");
    fgets(message, BUFFER_SIZE, stdin);

    write(fd, message, strlen(message) + 1);
    close(fd);

    fd = open(FIFO2, O_RDONLY);
    read(fd, message, BUFFER_SIZE);
    printf("User 1 received: %s", message);
    close(fd);
}

return 0;
}

```

Receiver:

```

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>

#define FIFO1 "fifo1"
#define FIFO2 "fifo2"
#define BUFFER_SIZE 1024

int main() {
    char message[BUFFER_SIZE];

```

```
int fd;

while (1) {

    fd = open(FIFO1, O_RDONLY);
    read(fd, message, BUFFER_SIZE);
    printf("User 2 received: %s", message);
    close(fd);

    fd = open(FIFO2, O_WRONLY);
    printf("User 2: ");
    fgets(message, BUFFER_SIZE, stdin);

    write(fd, message, strlen(message) + 1);
    close(fd);
}

return 0;
}
```



```
root@xit: /media/xit/3rd/University Tasks/FAS... x root@xit: /media/xit/3rd/University Tasks/FAS... x
xit@xit:/media/xit/3rd/University Tasks/FAST-BSE-5B/OS Lab/Lab_07$ sudo su
[sudo] password for xit:
root@xit:/media/xit/3rd/University Tasks/FAST-BSE-5B/OS Lab/Lab_07# ./t4-reciever
r
User 2 received:
User 2:
User 2 received:
User 2:
User 2 received: hi
User 2: Hi
User 2 received:
User 2: what are you doing
User 2 received:
User 2:
User 2 received:
User 2:
User 2 received: i am doing nothing
User 2: 
```

```
root@xit:/media/xit/3rd/University Tasks/FAST-BSE-5B/OS Lab/Lab_07# mkfifo fifo1
mkfifo fifo2
root@xit:/media/xit/3rd/University Tasks/FAST-BSE-5B/OS Lab/Lab_07# ./
t1          t4-reciever t4-sender  test       .vscode/
root@xit:/media/xit/3rd/University Tasks/FAST-BSE-5B/OS Lab/Lab_07# ./t
t1          t4-reciever t4-sender  test
root@xit:/media/xit/3rd/University Tasks/FAST-BSE-5B/OS Lab/Lab_07# ./t4-sender

User 1: hi
User 1 received:
User 1: User 1 received:
User 1: User 1 received: Hi
User 1:

User 1 received: what are you doing
User 1: i am doing nothing
User 1 received:
User 1: User 1 received:
User 1: 
```