

به نام خدا

گزارش آزمایش چهارم آزمایشگاه سیستم های عامل

زهرا رحیمی

شماره دانشجویی: ۹۸۳۱۰۲۶

استاد آزمایشگاه: سرکار خانم حسینی

پاییز ۱۴۰۰

بخش اول:

روش shared memory برای برقراری ارتباط:

در این روش ابتدا با استفاده از shmget، آیدی حافظه مشترک را ایجاد می کنیم، سپس با استفاده از shmat، و آن آیدی حافظه مشترک را ایجاد کرده و با ساختن فرآیند و نوشتن داده در حافظه مشترک در فرآیند فرزند، و خواندن آن در فرآیند پدر، هدف سوال را برآورده می کنیم.

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/shm.h>
#include <sys/stat.h>
#include <unistd.h>
#include <wait.h>
#include <fcntl.h>
int main ()
{
    int segment_id;
    char* shared_memory;
    struct shmid_ds shmbuffer;
    const int shared_segment_size = 0x6400;

    segment_id = shmget (IPC_PRIVATE, 1000, S_IRUSR | S_IWUSR);
    shared_memory = (char*) shmat(segment_id, NULL, 0);
    //shmctl (segment_id, IPC_STAT, &shmbuffer);
    pid_t pid = fork();

    if (pid < 0){
        printf("failed to create process");
        exit(EXIT_FAILURE);
    } else if (pid == 0){
        sprintf(shared_memory, "Writing to shared memory from process1\n");
        exit(EXIT_SUCCESS);
    } else {
        waitpid(pid, NULL, WUNTRACED | WCONTINUED);
        printf("process 2: \n");
        printf("Data read from shared memory: %s\n", shared_memory);
    }

    shmdt(shared_memory);
    shmctl (segment_id, IPC_RMID, 0);
    return 0;
}
```

خروجی:

```
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/part1$ gcc shmem.c -o shmem
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/part1$ ./shmem
process 2:
Data read from shared memory: Writing to shared memory from process1
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/part1$
```

بخش دوم:

از آنجا که کد نسبتاً طولانی است فقط فایل آن را در پیوست قرار می‌دهم. یک شبیه‌سازی نسبتاً ساده از خروجی را در زیر می‌بینیم که از تمام دستورات `join <group_id>`, `send <group_id>`, `<message>`, `leave<group_id>`, `quit` استفاده شده است:

بخش سرور:

```
zahra@zahra-virtual-machine: ~/Desktop/OS_Lab/az4/tmp3
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$ gcc server.c -pthread -o server
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$ ./server 4446
=== WELCOME TO THE CHATROOM ===
zahra has joined
fateme has joined
hamide has joined
fateme joined group with id 1
zahra joined group with id 2
hamide joined group with id 2
fateme joined group with id 3
zahra joined group with id 3
hamide joined group with id 1
salam hamide
sent from fateme to group 1
salam zahra
sent from hamide to group 1
khoobi fateme?
sent from zahra to group 3
merci zahra:)
sent from fateme to group 3
^C
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$
```

کلاینت اول:

```
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$ gcc client.c -pthread -o client
client.c: In function 'main':
client.c:81:1: warning: format not a string literal and no format arguments [-Wformat-security]
    sprintf(name, argv[3]);
    ~~~~~^
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$ ./client 127.0.0.1 4446 zahra
=== WELCOME TO THE CHATROOM ===
> join 2
> hamide has joined the group
> join 3
> send 3 khoobi fateme?
> fateme : merci zahra:)
> leave 3
> quit

Thank you for your support
Please come back soon
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$
```

کلاینت دوم:

```
zahra@zahra-virtual-machine: ~/Desktop/OS_Lab/az4/tmp3
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$ ./client 127.0.0.1 4446 f
ateme
=== WELCOME TO THE CHATROOM ===
> join 1
> join 3
> zahra has joined the group
> hamide has joined the group
> send 1 salam hamide
> hamide : salam zahra
> zahra : khoobi fateme?
> send 3 merci zahra:)
> leave 1
> hamide has left the group
> quit

Thank you for your support
Please come back soon
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$
```

کلاینت سوم:

```
zahra@zahra-virtual-machine: ~/Desktop/OS_Lab/az4/tmp3
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$ ./client 127.0.0.1 4446 h
amide
=== WELCOME TO THE CHATROOM ===
> join 2
> join 1
> fateme : salam hamide
> send 1 salam zahra
> fateme has left the group
> leave 1
> quit

Thank you for your support
Please come back soon
zahra@zahra-virtual-machine:~/Desktop/OS_Lab/az4/tmp3$
```

همانطور که مشاهده می کنیم سه فرد به درستی به گروه های دلخواه شان عضو شدند، پیام های افراد فقط برای افراد همان گروه فرستاده شده است و سپس از گروه خارج شده و از برنامه خارج شده اند.

بخش سوم:

این بخش با استفاده از دو خط لوله پیاده سازی شده است: به این صورت که خط لوله اول دیتا را از ورودی گرفته و به خط لوله دوم می دهد تا عملیات (کوچک کردن حروف بزرگ و بزرگ کردن حروف کوچک) را انجام دهد.

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<unistd.h>
4  #include<ctype.h>
5  #include <string.h>
6  #include<sys/wait.h>
7
8  int main() {
9      int fd1[2];
10     int fd2[2];
11     char input_str[100];
12     pid_t p;
13
14     if (pipe(fd1) == -1) {
15         fprintf(stderr, "Pipe Failed" );
16         return 1;
17     }
18     if (pipe(fd2) == -1) {
19         fprintf(stderr, "Pipe Failed" );
20         return 1;
21     }
22
23     gets(input_str);
24     p = fork();
25
26     if (p < 0) {
27         fprintf(stderr, "fork Failed" );
28         return 1;
29     } // Parent process
```

```

29 } // Parent process
30 else if (p > 0) {
31     char output_str[100];
32     close( fildes: fd1[0]);
33     write( fd: fd1[1], input_str, nbyte: strlen(input_str)+1);
34     close( fildes: fd1[1]);
35     wait( status: NULL);
36     close( fildes: fd2[1]);
37     read( fd: fd2[0], output_str, nbyte: 100);
38     printf("Output string is:\n%s\n", output_str);
39     close( fildes: fd2[0]);
40 } // child process
41 else {
42     close( fildes: fd1[1]);
43     char input[100];
44     read( fd: fd1[0], input, nbyte: 100);
45     char up_down_str[100];
46     int i;
47     for (i = 0; i < strlen(input); i++) {
48         if (isupper( _c: input[i])){
49             up_down_str[i] = tolower( _c: input[i]);
50         } else {
51             up_down_str[i] = toupper( _c: input[i]);
52         }
53     }
54     up_down_str[i] = '\0';
55     close( fildes: fd1[0]);
56     close( fildes: fd2[0]);
57     write( fd: fd2[1], up_down_str, nbyte: strlen(up_down_str)+1);
58     close( fildes: fd2[1]);
59     exit( status: 0);
60 }

```

خروجی به ازای رشته This Is First Process:

```

E:\uni\Term5\os\lab\4\part3\cmake-build-debug\part3.exe
This Is First Process
Output string is:
tHIS IS fIRST pROCESS

Process finished with exit code 0

```