

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Одеський національний політехнічний університет  
Інститут комп'ютерних систем  
Кафедра інформаційних систем

Звіт  
Лабораторної роботи № 12  
З предмету «Операційні системи»

**Тема: «Програмування міжпроцесної та багатопоточної взаємодії»**

Виконала:  
Студентка групи АІ-205  
Шаповалова Вікторія

Перевірили:  
Блажко О. А.  
Дрозд М. О.

Одеса 2021

**Мета роботи:** вивчити особливості обміну інформацією між процесами за допомогою іменованих каналів, керування потоками, а також синхронізацію процесів через семафори та мьютекси.

### **Вимоги до оформлення протоколу виконання лабораторної роботи**

Протокол у електронному вигляді повинен мати наступну структуру

- 1) титульний аркуш з назвою дисципліни, теми лабораторної роботи, групи та ПІБ виконавця;
- 2) аркуш із завданням до лабораторної роботи;
- 3) аркуші з результатами виконання пунктів завдань:
  - пункт із завданням;
  - скріншот частини екрану з командами ( з кольорами тексту білий фон/чорний тон);
  - скріншот частини екрану з результатом виконання команди;
- 4) аркуш з висновками:
  - перелік завдань, які були найскладнішими.

### **Завдання 1. Робота з іменованими каналами**

**1.1** В домашньому каталозі вашого користувача створіть іменований канал з використанням команди `mkfifo`:

- назва каналу співпадає з вашим прізвищем у транслітерації
- права доступу до каналу ( можна лише читати та писати власнику).

```
[shapovalova_viktoriya@vpsj3IeQ ~]$ mkfifo shapovalovav
[shapovalova_viktoriya@vpsj3IeQ ~]$ chmod u=rw,g=,o= shapovalovav
[shapovalova_viktoriya@vpsj3IeQ ~]$ ls -l
```

```
lrwxrwxrwx 1 shapovalova_viktoriya shapovalova_viktoriya 7 May 16 18:56 shap3.sh -> shap.sh
d-w--wxrw- 2 shapovalova_viktoriya shapovalova_viktoriya 4096 Apr 12 08:25 shapovalova
-rw-rw-r-- 1 shapovalova_viktoriya shapovalova_viktoriya 0 Mar 21 20:38 shapovalova_lab_3
drwxrwxr-x 2 shapovalova_viktoriya shapovalova_viktoriya 4096 Mar 22 09:42 shapovalova_lab_31
prw----- 1 shapovalova_viktoriya shapovalova_viktoriya 0 May 27 09:21 shapovalovav
-rw-rw-r-- 3 shapovalova_viktoriya shapovalova_viktoriya 0 Apr 13 20:15 shapovalova_viktoriya.txt
-rw-rw-r-- 1 shapovalova_viktoriya shapovalova_viktoriya 63 May 16 18:43 shap.sh
```

**1.2** Підключіть до іменованого каналу процес, який буде в нього писати за такими командами:

- отримати зміст каталогу /etc
- отримати назви файлів, які починаються з букви вашого прізвища у транслітерації.

**1.3** Перейдіть до нового терміналу роботи з ОС Linux та створіть процес, який буде читати зі створеного раніше каналу.

```
[shapovalova_viktoriya@vpsj3IeQ ~]$ ls /etc | grep "^s" > shapovalovav  
[shapovalova_viktoriya@vpsj3IeQ ~]$
```

```
login as: shapovalova_viktoriya  
shapovalova_viktoriya@91.219.60.189's password:  
Last login: Thu May 27 09:19:35 2021 from 213.231.44.202.pool.breezein.net  
[shapovalova_viktoriya@vpsj3IeQ ~]$ cat shapovalovav  
sas12  
securetty  
security  
selinux  
services  
sestatus.conf  
sgml  
shadow  
shadow-  
shells  
shells.rpmnew  
skel  
smartmontools  
smrsh  
ssh  
ssl  
statetab  
statetab.d  
subgid  
subgid-  
subuid  
subuid-  
sudo.conf  
sudoers  
sudoers.d  
sudo-ldap.conf  
sysconfig  
sysctl.conf  
sysctl.d  
systemd  
system-release  
system-release-cpe  
[shapovalova_viktoriya@vpsj3IeQ ~]$
```

**1.4** Поверніться до 1-го терміналу та підключіть до іменованого каналу процес, який буде в нього писати, архівуючи файл командою `gzip -c < pipe > file1.gz` де `pipe` – назва вашого каналу, `file1.gz` – назва файлу, який буде створено в результаті архівації

```
[shapovalova_viktoriya@vpsj3IeQ ~]$ gzip -c < shapovalovav > archive.gz
```

**1.5** Перейдіть до 2-го терміналу роботи з ОС Linux та створіть процес, який буде читати зі створеного раніше каналу, архівуючи файл /etc/passwd

```
[shapovalova_viktoriya@vpsj3IeQ ~]$ cat /etc/passwd > shapovalovav
[shapovalova_viktoriya@vpsj3IeQ ~]$ gunzip -c archive.gz
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:997:User for polkitd:/:/sbin/nologin
postfix:x:89:89:/:/var/spool/postfix:/sbin/nologin
chrony:x:998:996:/:/var/lib/chrony:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
soft:x:1000:1000:/:/home/soft:/sbin/nologin
saslauthd:x:997:76:Saslauthd user:/run/saslauthd:/sbin/nologin
mailnull:x:47:47:/:/var/spool/mqueue:/sbin/nologin
smmsp:x:51:51:/:/var/spool/mqueue:/sbin/nologin
emps:x:996:1001:/:/home/emps:/bin/bash
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
oracle:x:54321:54321:/:/home/oracle:/bin/bash
apache:x:995:995:/:/home/apache:/sbin/nologin
mysql:x:27:27:MySQL Server:/var/lib/mysql:/bin/false
named:x:25:25:Named:/var/named:/sbin/nologin
exim:x:93:93:/:/var/spool/exim:/sbin/nologin
vmmail:x:5000:5000:/:/var/local/vmail:/bin/bash
arestov_yaroslav:x:54322:54331:/:/home/arestov_yaroslav:/bin/bash
martinyuk_denis:x:54323:54332:/:/home/martinyuk_denis:/bin/bash
bogachik_egor:x:54324:54333:/:/home/bogachik_egor:/bin/bash
eliseev_danilo:x:54325:54334:/:/home/eliseev_danilo:/bin/bash
bochkarova_karina:x:54326:54335:/:/home/bochkarova_karina:/bin/bash
tsigankova_anastasiya:x:54327:54327:/:/home/tsigankova_anastasiya:/bin/bash
vyatkin_danilo:x:54328:54328:/:/home/vyatkin_danilo:/bin/bash
harkevich_leonid:x:54329:54329:/:/home/harkevich_leonid:/bin/bash
kalarashan_kostyantyn:x:54330:54336:/:/home/kalarashan_kostyantyn:/bin/bash
nosov_andrij:x:54331:54337:/:/home/nosov_andrij:/bin/bash
kebab_dmitro:x:54332:54338:/:/home/kebab_dmitro:/bin/bash
prihodko_egor:x:54333:54339:/:/home/prihodko_egor:/bin/bash
kelembet_evgen:x:54334:54340:/:/home/kelembet_evgen:/bin/bash
malofeev_denis:x:54335:54341:/:/home/malofeev_denis:/bin/bash
tarasov_daniil:x:54336:54342:/:/home/tarasov_daniil:/bin/bash
stepanenko_gleb:x:54337:54343:/:/home/stepanenko_gleb:/bin/bash
strinaglyuk_ivan:x:54338:54344:/:/home/strinaglyuk_ivan:/bin/bash
yufitenko_andrij:x:54339:54345:/:/home/yufitenko_andrij:/bin/bash
cheremnih_grigoriy:x:54340:54346:/:/home/cheremnih_grigoriy:/bin/bash
barkar_karina:x:54341:54347:/:/home/barkar_karina:/bin/bash
loboda_oleksandra:x:54342:54348:/:/home/loboda_oleksandra:/bin/bash
borovets_vladislav:x:54343:54349:/:/home/borovets_vladislav:/bin/bash
matnenko_stanislav:x:54344:54350:/:/home/matnenko_stanislav:/bin/bash
bodnar_illya:x:54345:54351:/:/home/bodnar_illya:/bin/bash
nezhivih_mariya:x:54346:54352:/:/home/nezhivih_mariya:/bin/bash
```

## **Завдання 2 Програмування іменованих каналів**

Повторіть попереднє завдання, але пункт 1.1 виконайте через програмування іменованого каналу за прикладом з рисунку 1.

```
shapovalova_viktoriya@vpsj3IeQ:~  
GNU nano 2.3.1 File: shapovalova2.c  
  
#include <sys/stat.h>  
#include <fcntl.h>  
#include <string.h>  
#include <stdio.h>  
  
#define NAMEDPIPE_NAME "shapovalova2"  
#define BUFSIZE 50  
  
int main (int argc, char ** argv) {  
    int fd, len;  
    char buf[BUFSIZE];  
  
    if ( mkfifo(NAMEDPIPE_NAME, 0777) ) {  
        fprintf(stderr, "Error in mkfifo!");  
        return 1;  
    }  
    printf("%s is created\n", NAMEDPIPE_NAME);  
  
    if ( (fd = open(NAMEDPIPE_NAME, O_RDONLY)) <= 0 ) {  
        fprintf(stderr, "Error in open!");  
        return 1;  
    }  
    printf("%s is opened\n", NAMEDPIPE_NAME);  
  
    do {  
        memset(buf, '\0', BUFSIZE);  
        if ( (len = read(fd, buf, BUFSIZE-1)) <= 0 ) {  
            printf("END!");  
            close(fd);  
            remove(NAMEDPIPE_NAME);  
            return 0;  
        }  
        printf("Incomming message (%d): %s\n", len, buf);  
    } while ( 1 );  
}
```

```
[shapovalova_viktoriya@vpsj3IeQ ~]$ gcc shapovalova2.c -o shapovalova2_  
[shapovalova_viktoriya@vpsj3IeQ ~]$ ./shapovalova2_  
shapovalova2 is created  
shapovalova2 is opened  
END! [shapovalova_viktoriya@vpsj3IeQ ~]$
```

### Завдання 3 Програмування потоків

За прикладом з рисунку 2 розробіть програму керування потоками, в якій в повідомленнях буде вказано ваше прізвище латиницею.

Виконайте програму за вказаним прикладом.

```
shapovalova_viktoriya@vpsj3IeQ:~  
GNU nano 2.3.1 File: sha  
#include <stdio.h>  
#include <pthread.h>  
  
main() {  
    pthread_t f2_thread, f1_thread;  
    void *f2(), *f1();  
    int i1 = 10, i2 = 10;  
    pthread_create(&f1_thread, NULL, f1, &i1);  
    pthread_create(&f2_thread, NULL, f2, &i2);  
    pthread_join(f1_thread, NULL);  
    pthread_join(f2_thread, NULL);  
}  
  
void *f1(int *x) {  
    int i,n;  
    n = *x;  
    for (i=1;i<n;i++) {  
        printf("Shapovalova_1: %d\n", i);  
        sleep(1);  
    }  
    pthread_exit(0);  
}  
  
void *f2(int *x) {  
    int i,n;  
    n = *x;  
    for (i=1;i<n;i++) {  
        printf("Shapovalova_2: %d\n", i);  
        sleep(1);  
    }  
    pthread_exit(0);  
}
```

```
[shapovalova_viktoriya@vpsj3IeQ ~]$ gcc -pthread shapovalova3.c -o shapovalova3  
[shapovalova_viktoriya@vpsj3IeQ ~]$ ./shapovalova3  
Shapovalova_1: 1  
Shapovalova_2: 1  
Shapovalova_1: 2  
Shapovalova_2: 2  
Shapovalova_1: 3  
Shapovalova_2: 3  
Shapovalova_1: 4  
Shapovalova_2: 4  
Shapovalova_1: 5  
Shapovalova_2: 5  
Shapovalova_1: 6  
Shapovalova_2: 6  
Shapovalova_1: 7  
Shapovalova_2: 7  
Shapovalova_1: 8  
Shapovalova_2: 8  
Shapovalova_1: 9  
Shapovalova_2: 9  
[shapovalova_viktoriya@vpsj3IeQ ~]$
```

#### Завдання 4 Програмування семафорів

За прикладом з рисунку 3 розробіть програму керування семафором, в якій в повідомленнях буде вказано ваше прізвище латиницею. Виконайте програму в двох терміналах за вказаним прикладом.

```
GNU nano 2.3.1 File: shapovalovasem.c

include <fcntl.h>
include <sys/stat.h>
include <semaphore.h>
include <stdio.h>

define SEMAPHORE_NAME "/shapovalova_sem"

nt main(int argc, char ** argv) {

    sem_t *sem;

    if ( argc != 2 ) {
        if ((sem = sem_open(SEMAPHORE_NAME, O_CREAT, 0777, 0)) == SEM_FAILED ) {
            fprintf(stderr,"Shapovalova sem_open error");
            return 1;
        }
        printf("Shapovalova sem_open. Semaphore is taken.\nWaiting for it to be dropped.\n");
        if (sem_wait(sem) < 0 )
            fprintf(stderr,"Shapovalova sem_wait error");
        if ( sem_close(sem) < 0 )
            fprintf(stderr,"Shapovalova sem_close error");
        return 0;
    }
    else {
        printf("Dropping Shapovalova semaphore...\n");
        if ( (sem = sem_open(SEMAPHORE_NAME, 0)) == SEM_FAILED ) {
            fprintf(stderr,"Shapovalova sem_open error");
            return 1;
        }
        sem_post(sem);
        printf("Shapovalova sem_post. Semaphore dropped.\n");
        return 0;
    }
}
```

```
Shapovalova sem_open error[shapovalova_viktoriya@vpsj3IeQ ~]$ gcc -pthread shapovalovasem.c -o shapovalovasem
[shapovalova_viktoriya@vpsj3IeQ ~]$ ./shapovalovasem 1
Dropping Shapovalova semaphore...
Shapovalova sem_post. Semaphore dropped.
[shapovalova_viktoriya@vpsj3IeQ ~]$
```

```
collectd: error: ld returned 1 exit status
[shapovalova_viktoriya@vpsj3IeQ ~]$ gcc -pthread shapovalovasem.c -o shapovalovasem
[shapovalova_viktoriya@vpsj3IeQ ~]$ ./shapovalovasem
Shapovalova sem_open. Semaphore is taken.
Waiting for it to be dropped.
[shapovalova_viktoriya@vpsj3IeQ ~]$
```

**Висновок:** в ході роботи ми вивчили особливості обміну інформацією між процесами за допомогою іменованих каналів, керування потоками, а також синхронізацію процесів через семафори та мьютекси. Найскладнішою була теорія.