

Министерство образования Республики Беларусь Учреждение
образования
«Брестский государственный технический университет» Кафедра
ИИТ

Лабораторная работа №7
по дисциплине «Операционные системы и системное программирование»
Тема: «Семафоры»
Вариант 5

Выполнил: студент
2-го курса группы ПО-
6 Лавренчик
Д.О.
Проверил: Давидюк
Ю.И.

Брест 2022
Лабораторная работа №7

Ход работы

Задание для выполнения:

Вариант 5. Первый процесс в цикле ожидает ввода символа с потока stdin, после чего пишет в файл случайное число, каждый раз открывая и закрывая за собой файл. Второй процесс забирает из файла числа и выводит на экран соответствующее числу количество любых символов.

Код программы:

First.cpp:

```
#include <sys/types.h>
#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>
#include <semaphore.h>
#include <stdlib.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <iostream>
#include <cstring>
#include <sys/stat.h>
#include <ctime>
#include <pthread.h>

int main() {
    int fd = 0;
    srand(time(0));
    char str[255];
    char enter[] = "Enter the symbol: ";
    sem_t* f_semaph = sem_open("sem1", O_CREAT, 0777, 0);
    sem_t* s_semaph = sem_open("sem2", O_CREAT, 0777, 0);
    sem_post(s_semaph);
    (void)umask(0);

    while (1) {
        sem_wait(s_semaph);
        write(0, enter, strlen(enter));
        read(0, str, 255);
        printf("%d", getpid());

        pthread_mutex_t mutex = PTHREAD_MUTEX_INITIALIZER;
        pthread_mutex_lock(&mutex);
        fd = open("temp.txt", O_WRONLY | O_CREAT | O_TRUNC, 0777);
        int num = 1 + rand() % 10;
        printf(" sends symbol: %d\n", num);
        write(fd, &num, 1);
        close(fd);
        pthread_mutex_unlock(&mutex);
        pthread_mutex_destroy(&mutex);
        sem_post(f_semaph);
    }
}
```

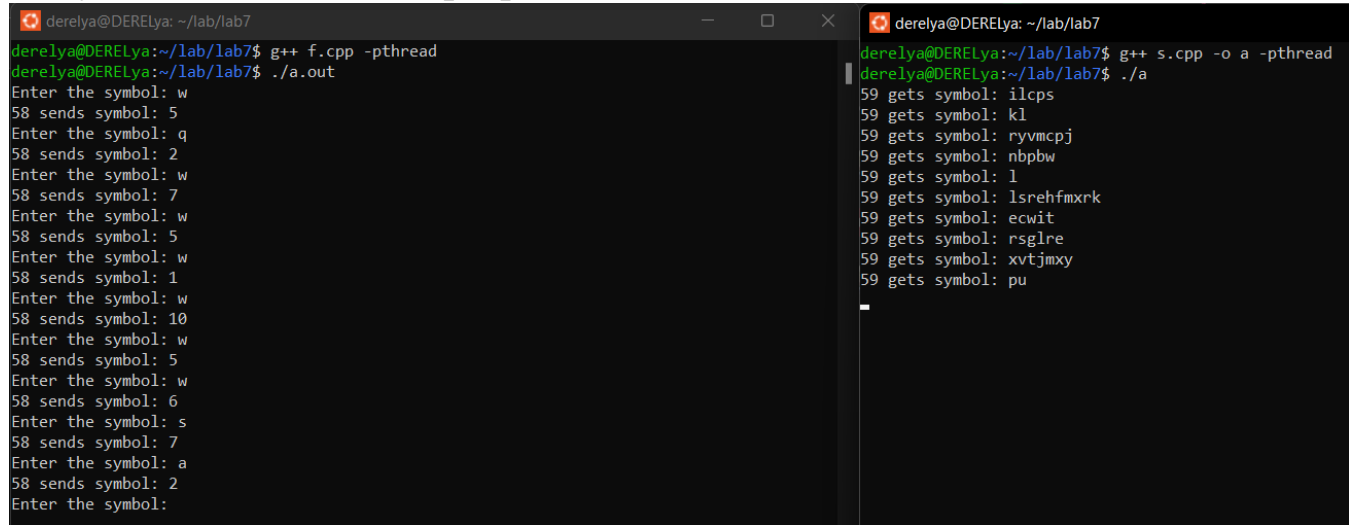
```
}
```

```
sem_close(f_semaph);  
sem_close(s_semaph);  
return 0;
```

}Second.cpp:

```
#include <unistd.h>  
#include <sys/types.h>  
#include <semaphore.h>  
#include <stdio.h>  
#include <fcntl.h>  
#include <stdlib.h>  
#include <sys/ipc.h>  
#include <sys/shm.h>  
#include <pthread.h>  
int main() {  
    int fd = 0;  
    char symb;  
    sem_t* f_semaph = sem_open("sem1", O_CREAT, 0777, 0);  
    sem_t* s_semaph = sem_open("sem2", O_CREAT, 0777, 0);  
    pthread_mutex_t mutex = PTHREAD_MUTEX_INITIALIZER;  
    while (1) {  
        sem_wait(f_semaph);  
        pthread_mutex_lock(&mutex);  
        fd = open("temp.txt", O_RDWR);  
        int num = 0;  
        read(fd, &num, 1);  
        printf("%d", getpid());  
        printf(" gets symbol: ");  
        for (int i = 0; i < num; ++i) {  
            symb = 97 + rand() % 25;  
            printf("%c", symb);  
        }  
        printf("\n");  
        close(fd);  
        pthread_mutex_unlock(&mutex);  
        sem_post(s_semaph);  
    }  
    sem_close(f_semaph);  
    sem_close(s_semaph);  
    pthread_mutex_destroy(&mutex);  
    return 0;
```

Результат выполнения программы:



```
derelya@DERELya: ~/lab/lab7
derelya@DERELya:~/lab/lab7$ g++ f.cpp -pthread
derelya@DERELya:~/lab/lab7$ ./a.out
Enter the symbol: w
58 sends symbol: 5
Enter the symbol: q
58 sends symbol: 2
Enter the symbol: w
58 sends symbol: 7
Enter the symbol: w
58 sends symbol: 5
Enter the symbol: w
58 sends symbol: 1
Enter the symbol: w
58 sends symbol: 10
Enter the symbol: w
58 sends symbol: 5
Enter the symbol: w
58 sends symbol: 6
Enter the symbol: s
58 sends symbol: 7
Enter the symbol: a
58 sends symbol: 2
Enter the symbol:

derelya@DERELya: ~/lab/lab7
derelya@DERELya:~/lab/lab7$ g++ s.cpp -o a -pthread
derelya@DERELya:~/lab/lab7$ ./a
59 gets symbol: ilcps
59 gets symbol: kl
59 gets symbol: ryvmcpj
59 gets symbol: nbpbw
59 gets symbol: l
59 gets symbol: lsrehfmxrk
59 gets symbol: ecwit
59 gets symbol: rsglre
59 gets symbol: xvtjmxxy
59 gets symbol: pu
```

Вывод: в ходе выполнения данной лабораторной работы были изучены семафоры в операционной системе UNIX.