

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”
КАФЕДРА ИИТ

ОТЧЁТ
по лабораторной работе №7
«Семафоры»

Выполнил:

Студент 2 курса
группы ПО-9
Мисиюк Алексей Сергеевич
(№ зач. книги 210664)

Проверила:

Давидюк Ю. И.

Брест 2022

Цель работы: изучить и применить в работе семафоры в ОС Линукс.

Ход работы

Вариант индивидуального задания № 13

Первый процесс в цикле ожидает ввода символа с потока stdin, после чего пишет в файл соответствующий символ, каждый раз открывая и закрывая за собой файл. Второй процесс забирает из файла символы и выводит на экран только гласные буквы.

Код программы (Производитель)

```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <pthread.h>
#include <semaphore.h>

const char empty_name[] = "report7_sem_empty";
const char full_name[] = "report7_sem_full";

pthread_mutex_t mutex;

int main() {

    sem_t* empty = sem_open(empty_name, O_CREAT, 0644, 1);
    sem_t* full = sem_open(full_name, O_CREAT, 0644, 0);
    pthread_mutex_init(&mutex, NULL);

    // model of a Producer processor

    while (1) {

        sem_wait(empty); //P(empty)

        char chr;
        if (!read(0, &chr, 1)) {
            continue;
        }

        pthread_mutex_lock(&mutex);

        //critical section
        int fd;
        if ((fd = open("file.tmp", O_CREAT | O_RDWR, 0666)) < 0) {
            perror("Error opening file!");
```

```

        exit(-1);
    }

    write(fd, &chr, 1);

    close(fd);
    //critical section end

    pthread_mutex_unlock(&mutex);

    sem_post(full); //V(full)
}

return 0;
}

```

Код программы (Потребитель)

```

#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <pthread.h>
#include <semaphore.h>
#include <string.h>

const char empty_name[] = "report7_sem_empty";
const char full_name[] = "report7_sem_full";

pthread_mutex_t mutex;

int main() {

    sem_t* empty = sem_open(empty_name, O_CREAT, 0644, 1);
    sem_t* full = sem_open(full_name, O_CREAT, 0644, 0);
    pthread_mutex_init(&mutex, NULL);

    char vowels[12] = "AEIOUYaeiouy";

    // model of a Consumer processor

    while (1) {
        char chr;

        sem_wait(full); //P(full)
    }
}

```

```

pthread_mutex_lock(&mutex);
//critical section
int fd;
if ((fd = open("file.tmp", O_RDWR, 0666)) < 0) {
    perror("Error opening file!");
    exit(-1);
}

read(fd, &chr, 1);

close(fd);
//critical section end
pthread_mutex_unlock(&mutex);

sem_post(empty); //V(empty)

if (strchr(vowels, chr) != NULL) {
    write(1, &chr, 1);
    chr = '\n';
    write(1, &chr, 1);
}
}

return 0;
}

```

Пример работы

Производитель 1:
qwertyuiopasdfghjkl

Потребитель 1:

е

и

о

Потребитель 2:

у

і

а

Вывод: семафоры и мьютексы являются довольно удобным инструментом для разграничения работы с критическими секциями.