

CS302: Lab8 Report

Name: 陆荻芸 SID: 12011537

Answer 1

Use a mutex lock before and after the critical section of mom and dad.

```

void *dad(){
    int fd;
    printf("Dad comes home.\n");
    sleep(rand()%2+1);
    pthread_mutex_lock(&mutex);
    printf("Dad checks the fridge.\n");
    fd=open("fridge", O_CREAT|O_RDWR|O_APPEND, 0777);
    if(lseek(fd,0,SEEK_END)!=0){
        printf("Dad goes to buy milk...\n");
        //sleep(rand()%2+1);
        printf("Dad comes back.\n");
        if(lseek(fd,0,SEEK_END)!=0){
            printf("What a waste of food! The fridge can not hold so much milk!\n");
        }
        else{
            write(fd,"milk",4);
            printf("Dad puts milk in fridge and leaves.\n");
            pthread_mutex_unlock(&mutex);
        }
    }
    printf("Dad closes the fridge and leaves.\n");
    close(fd);
}

pthread_mutex_t mutex;
void *mom(){
    int fd;
    printf("Mom comes home.\n");
    sleep(rand()%2+1);
    pthread_mutex_lock(&mutex);
    printf("Mom checks the fridge.\n");
    fd=open("fridge", O_CREAT|O_RDWR|O_APPEND, 0777);
    if(lseek(fd,0,SEEK_END)!=0){
        printf("Mom goes to buy milk...\n");
        //sleep(rand()%2+1);
        printf("Mom comes back.\n");
        if(lseek(fd,0,SEEK_END)!=0){
            printf("What a waste of food! The fridge can not hold so much milk!\n");
        }
        else{
            write(fd,"milk",4);
            printf("Mom puts milk in fridge and leaves.\n");
            pthread_mutex_unlock(&mutex);
        }
    }
    printf("Mom closes the fridge and leaves.\n");
    close(fd);
}

* (base) ldy12011537@lady12011537:~/Desktop/05Lab8/4/lab8: ./milk
Mom comes home.
Dad comes home.
Dad checks the fridge.
Dad goes to buy milk...
Dad comes back.
Dad puts milk in fridge and leaves.
Mom checks the fridge.
Mom closes the fridge and leaves.

```

(a) Dad

(b) Mom

(c) Result

图 1: Answer 1

Answer 2

Mom and sister are buyers who put 5 milks in the fridge when the fridge is empty. You and dad are consumers, drink one milk each time and wait when fridge is empty.

```

void* mom(){
    // int i=5;
    while (1)
    {
        pthread_mutex_lock(&mutex);
        printf("Mom locks\n");
        while (milk > 0)
        {
            printf("Mom goes to wait\n");
            pthread_cond_wait(&cond, &mutex);
        }
        printf("Mom goes to buy milk...\n");
        milk+=5;
        printf("Mom unlocks\n");
        pthread_mutex_unlock(&mutex);
    }
}

void* sister()
{
    while (1)
    {
        pthread_mutex_lock(&mutex);
        printf("Sister locks\n");
        while (milk > 0)
        {
            printf("Sister goes to wait\n");
            pthread_cond_wait(&cond, &mutex);
        }
        printf("Sister goes to buy milk...\n");
        milk+=5;
        printf("Sister unlocks\n");
        pthread_mutex_unlock(&mutex);
    }
}

void* dad(){
    while(1){
        pthread_mutex_lock(&mutex);
        printf("Dad locks\n");
        if(milk>0){
            milk--;
            printf("Dad drinks milk...\n");
        }else{
            pthread_cond_signal(&cond);
            printf("Dad waits for milk...\n");
        }
        pthread_mutex_unlock(&mutex);
        printf("Dad unlocks\n");
    }
}

void* you()
{
    // int i=5;
    while(1){
        pthread_mutex_lock(&mutex);
        printf("You lock\n");
        if(milk>0){
            milk--;
            printf("You drink milk...\n");
        }else{
            pthread_cond_signal(&cond);
            printf("You wait for milk\n");
        }
        pthread_mutex_unlock(&mutex);
        printf("You unlock\n");
    }
}

```

(a) mom

(b) sister

(c) dad

(d) you

图 2: Answer 2

Result shown as below:

```
Dad locks
Dad drinks milk...
Dad unlocks
Dad locks
Dad waits for milk...
Dad unlocks
Mom goes to buy milk...
Mom unlocks
Mom locks
Mom goes to wait
Dad locks
Dad drinks milk...
Dad unlocks
Dad locks
Dad drinks milk...
Dad unlocks
You lock
You drink milk...
You unlock
You lock
You drink milk...
You unlock
You lock
You drink milk...
You unlock
You lock
You wait for milk
Sister goes to buy milk...
Sister unlocks
```

图 3: Part of the results involving all members