Operation System Homework #2: Process Concept

0016046 蔡佩珊

PART A

A.1: FIFO version Code

(see attaced file: 0016046_FIFO.c)

A.2: SHM version Code

(see attaced file: 0016046_SHM.c)

A.3: Report

buff[3]/buff[4] records # of movements of Parent/Child pushed in SHM.

One of them values 1 while the other values 0.

Wait for the other side movement count being 1 to pop the movement from SHM.

Wait for their own side movement count being 0 to push the movement to SHM.

```
int shm;
int* buff;
buff=(int*)shmat(shm,0,0);
/* buff[0]=x-axis
   buff[1]=y-axis
   buff[2]=W-Flag
   buff[3]=Parent Movement Count
   buff[4]=Child  Movement Count */

/* Child Process */
while(buff[3]==0);// Wait for Parent Moving
...               // Pop Parent Movement
buff[3]--;        // Decrease Parent Movement Count

while(buff[4]==1);// Wait for Parent Reading Child Movement
...               // Push Child Movement
buff[4]++;        // Increase Child Movement Count

/* Parent Process */
while(buff[4]==0);// Wait for Child Moving
...               // Pop Child Movement
buff[4]--;        // Decrease Child Movement Count

while(buff[3]==1);// Wait for Child Reading Parent Movement
...               // Push Parent Movement
buff[3]++;        // Increase Parent Movement Count
```

PART B

B.1: What is the purpose of the program? What are the meanings of the output messages?

The purpose of the program is filling a block of memory with an integer in 4 different ways. The output messages are the duration time between the loop *for(int k=0; k<ROUND; k++) { ... }*

B.2: Tweak BUF_SIZE (in shared_mem.cpp) to 1024*1024*60 and redo the experiments. Describe your findings and explain the cause.

-DSHM, that is Using System V shared memory segment, crashed.

The query size is too large so that it is out of shared memory.

B.3: What are the technical differences between using -DSINGLE and -DCOPY?

Duplicate the BUF_X array or not.

If we try to write the same space (using -DSINGLE), the compiler may optimize the code and reduce half of assignments.

```
volatile int *BUF_X;
volatile int *BUF_Y;
BUF_X = new int[BUF_SIZE];

/* -DSINGLE */
BUF_Y = BUF_X;
/* -DCOPY */
BUF_Y = new int[BUF_SIZE];

for(int k=0; k<ROUND; k++)
{   /* -DSINGLE */
    Ping(BUF_X, -1);
    Ping(BUF_Y,  0);
    /* -DCOPY */
    Ping(BUF_X, -1);
    memcpy((void*)BUF_Y, (void*)BUF_X, BUF_SIZE*sizeof(int));
    Ping(BUF_Y,  0);
    memcpy((void*)BUF_X, (void*)BUF_Y, BUF_SIZE*sizeof(int));
}
```

B.4: What are the technical differences between using -DSINGLE and -DSHM?

Do *fork()* or not. Use single process or dual process.

```
/* -DSINGLE */
void SingleProcess()
{
    for(int k=0; k<ROUND; k++)
    {Ping(BUF_X, -1);
     Ping(BUF_Y,  0);}
}

/* -DSHM */
void TwoProcess()
{
    pid_t pidY;
    pidY = fork();

    if(pidY==0)
        for(int k=0; k<ROUND; k++)
            Ping(BUF_Y, 0);
    else
        for(int k=0; k<ROUND; k++)
            Ping(BUF_X,-1);
}
```

B.5: What are the technical differences between using -DSHM and -DMMAP?

Use system V APIs: *shmget /shmat / shmdt*

or POSIX APIs: *shm_open / mmap / munmap / shm_unlink*

The implementation of these two APIs are different, so the limitation of SHM size are different and cause different results of Experiment B.2.

Appendix

Experiment B

```
[demore@demore 2013_NCTU_OS_hw-master]$ g++ -O3 -DSINGLE ./shared_mem.cpp -lrt
[demore@demore 2013_NCTU_OS_hw-master]$ ./a.out

Single process mode

Duration : 17.7874 seconds
[demore@demore 2013_NCTU_OS_hw-master]$ g++ -O3 -DCOPY ./shared_mem.cpp -lrt
[demore@demore 2013_NCTU_OS_hw-master]$ ./a.out

Single process mode with data copy

Duration : 37.9989 seconds
[demore@demore 2013_NCTU_OS_hw-master]$ g++ -O3 -DSHM ./shared_mem.cpp -lrt
[demore@demore 2013_NCTU_OS_hw-master]$ ./a.out

Dual process mode

SHM_SIZE = 25165824
Using System V shared memory segment
BUF_X at 0x7fcd4c998000
BUF_Y at 0x7fcd4b198000
Duration : 19.0242 seconds
[demore@demore 2013_NCTU_OS_hw-master]$ g++ -O3 -DMMAP ./shared_mem.cpp -lrt
[demore@demore 2013_NCTU_OS_hw-master]$ ./a.out

Dual process mode

SHM_SIZE = 25165824
Using mmap
BUF_X at 0x7f46f5d12000
BUF_Y at 0x7f46f4512000
Duration : 21.681 seconds
```

Experiment B.2

BUF_SIZE=1024*1024*60

```
[demore@demore 2013_NCTU_OS_hw-master]$ g++ -O3 -DSINGLE ./shared_mem.cpp -lrt
[demore@demore 2013_NCTU_OS_hw-master]$ ./a.out

Single process mode

Duration : 193.045 seconds
[demore@demore 2013_NCTU_OS_hw-master]$ g++ -O3 -DCOPY ./shared_mem.cpp -lrt
[demore@demore 2013_NCTU_OS_hw-master]$ ./a.out

Single process mode with data copy

Duration : 453.482 seconds
[demore@demore 2013_NCTU_OS_hw-master]$ g++ -O3 -DSHM ./shared_mem.cpp -lrt
[demore@demore 2013_NCTU_OS_hw-master]$ ./a.out

Dual process mode

SHM_SIZE = 251658240
Using System V shared memory segment
a.out: ./shared_mem.cpp:144: void TwoProcess(): Assertion `sid !=-1' failed.
Aborted (core dumped)
[demore@demore 2013_NCTU_OS_hw-master]$ g++ -O3 -DMMAP ./shared_mem.cpp -lrt
[demore@demore 2013_NCTU_OS_hw-master]$ ./a.out

Dual process mode

SHM_SIZE = 251658240
Using mmap
BUF_X at 0x7fe72a433000
BUF_Y at 0x7fe71b433000
Duration : 240.274 seconds
```