

Programming Assignment 3 Part B

Binghan Geng A20482350

bgeng1@hawk.iit.edu

Yu Li A20496405

yli385@hawk.iit.edu

Introduction

This part is to learn how to implement a pair of xv6 system calls: `GetSharedPage()` and `FreeSharedPage()` that will allow two programs (two processes) to share pages.

Build & Run Instructions

compile: `make clean`
`make qemu-nox`

```
ubuntu@VM-176-15-ubuntu:~/xv6-public-xv6-rev9$ make qemu-nox
dd if=/dev/zero of=xv6.img count=10000
10000+0 records in
10000+0 records out
5120000 bytes (5.1 MB, 4.9 MiB) copied, 0.0293144 s, 175 MB/s
dd if=bootblock of=xv6.img conv=notrunc
1+0 records in
1+0 records out
512 bytes copied, 8.1483e-05 s, 6.3 MB/s
dd if=kernel of=xv6.img seek=1 conv=notrunc
363+0 records in
363+0 records out
185856 bytes (186 kB, 182 KiB) copied, 0.000429787 s, 432 MB/s
qemu-system-i386 -nographic -drive file=fs.img,index=1,media=disk,format=raw -drive file=xv6.img,index=0,media=disk,format=raw -smp 2 -m 512
xv6...
cpu1: starting
cpu0: starting
sb: size 1000 nblocks 941 ninodes 200 nlog 30 logstart 2          inodestart 32 bmap start 58
init: starting sh
$
```

execute syscall: `GetSharedPage` and `FreeSharedPage`

```
xv6...
cpu1: starting
cpu0: starting
sb: size 1000 nblocks 941 ninodes 200 nlog 30 logstart 2          inodestart 32 bmap start 58
init: starting sh
$ GetSharedPage
process pid:3
start write shared memory
return: key:1, address: 7FFFD000
write [Hello, CS450 PA3!] into key[1]-[7FFFD000]
$
```

```

process pid:3
start write shared memory
return: key:1, address: 7FFFD000
write [Hello,CS450 PA3!]into key[1]-[7FFFD000]
$ FreeSharedPage
process pid:4
start read shared memory
return: key:1, address: 7FFFD000
start release shared memory
FreeSharedPage: key is 1
FreeSharedPage: refcount is 2
FreeSharedPage: page_nums is 3
Free the user space memory.
free shared memory: key[1]-[7FFFD000]

```

execute Test Case: my_shell 3 3

```

Zombie      2 17 12430
GetSharedPage 2 18 12912
GetShmByParam 2 19 13140
FreeSharedPage 2 20 13072
FreeShmByParam 2 21 13380
my_shell     2 22 24020
console      3 23 0
$ my_shell 3 3
process pid:4
param: key:3, num_pages: dx
begin: share a memory page, key:3
return: key:3, address: 7FFFD000
write [Hello,XV6!]into key[3]-[7FFFD000]
CS450$ GetShmByParam 3 3

```

GetShmByParam 3 3

```

CS450$ GetShmByParam 3 3
process pid:5
start write shared memory
param: key:3, num_pages: dx
return: key:3, address: 7FFFD000
write [Hello,CS450 PA3!]into key[3]-[7FFFD000]
CS450$ FreeShmByParam 3 3

```

FreeShmByParam 3 3

```

write [Hello,CS450 PA3!]into key[3]-[7FFFD000]
CS450$ FreeShmByParam 3 3
process pid:6
start read shared memory
param: key:3, num_pages: dx
GetSharedPage Return: key:3, address: 7FFFD000
read [Hello,CS450 PA3!] from key[3]-[7FFFD000]
start release shared memory
FreeSharedPage: key is 3
FreeSharedPage: refcount is 3
FreeSharedPage: page_nums is 3
Free the user space memory.
free shared memory: key[3]-[7FFFD000]
CS450$

```