**Prerequisite:** git clone git://github.com/mit-pdos/xv6-public.git xv6

$ cd xv6

….xv6 $

Follow the below steps in Skill 9 folder

**Step1: Ps.c**

#include "types.h"

#include "stat.h"

#include "user.h"

#include "uproc.h"

#define STDOUT 1

#define NPROC 3 // maximum number of processes

void getprocdata()

{

int i = 0;

struct uproc up;

printf(STDOUT, "Name\tpid\tppid\tsize\t\twaiting\_on\_channel\tkilled\n");

printf(STDOUT, "-----------------------------------------------------------------------------\n");

for (i=0; i<NPROC; i++) {

getprocinfo(i, &up);

printf(STDOUT, "%s\t%d\t%d\t%d\t\t%d\t\t%d\n", up.name, up.pid, up.ppid, up.sz, up.wait, up.killed);

}

exit();

}

int

main(int argc, char \*argv[])

{

getprocdata();

exit();

}

**Step 2: Syscall.c**

extern int sys\_getprocinfo(void);

[SYS\_getprocinfo] sys\_getprocinfo,

**Step 3: Syscall.h**

#define SYS\_getprocinfo 22

**Step 4: Sysproc.c**

int

sys\_getprocinfo()

{

int proc\_num, size = sizeof(struct uproc);

struct uproc \*up;

if ((argint(0, &proc\_num) < 0) || (argptr(1, (char \*\*)&up, size) < 0))

return -1;

return getprocinfo(proc\_num, up);

}

**Step 5: User.h**

struct stat;

struct rtcdate;

**struct uproc;**

//Add this in System Calls block

**int getprocinfo(int, struct uproc\*);**

**Step 6: usys.S**

SYSCALL(getprocinfo)

**Step 7: Proc.h**

**//Add at the end**

struct uproc{

char name[16];

int pid;

int ppid;

uint sz;

int state;

int wait;

int killed;

};

**Step 8: Proc.c**

**// Add below code in proc.c at the end**

int

getprocinfo(int proc\_num, struct uproc \*up)

{

struct proc \*p;

if (proc\_num >= NPROC)

return -1;

p = &ptable.proc[proc\_num];

memset(up, 0, sizeof(struct uproc));

memmove(up->name, p->name, 16);

up->pid = p->pid;

up->state = p->state;

if ((up->state != UNUSED)) {

up->ppid = p->parent->pid;

up->sz = p->sz;

}

if (up->pid == 1) {

up->sz = p->sz;

up->ppid = 0;

}

if (p->chan)

up->wait = 1;

else

up->wait = 0;

up->killed = p->killed;

return 0;

}

**Step 9: defs.h**

**//Add uproc here..**

struct buf;

struct context;

struct file;

struct inode;

struct pipe;

struct proc;

**struct uproc;**

**//pagebreak 16 //proc.c (In defs.h)**

**int getprocinfo(int, struct uproc\*);**

**Step 10: uproc.h**

#include "types.h"

struct uproc{

char name[16];

int pid;

int ppid;

uint sz;

int state;

int wait;

int killed;

};

**Step 11: Makefile:**

$ nano Makefile

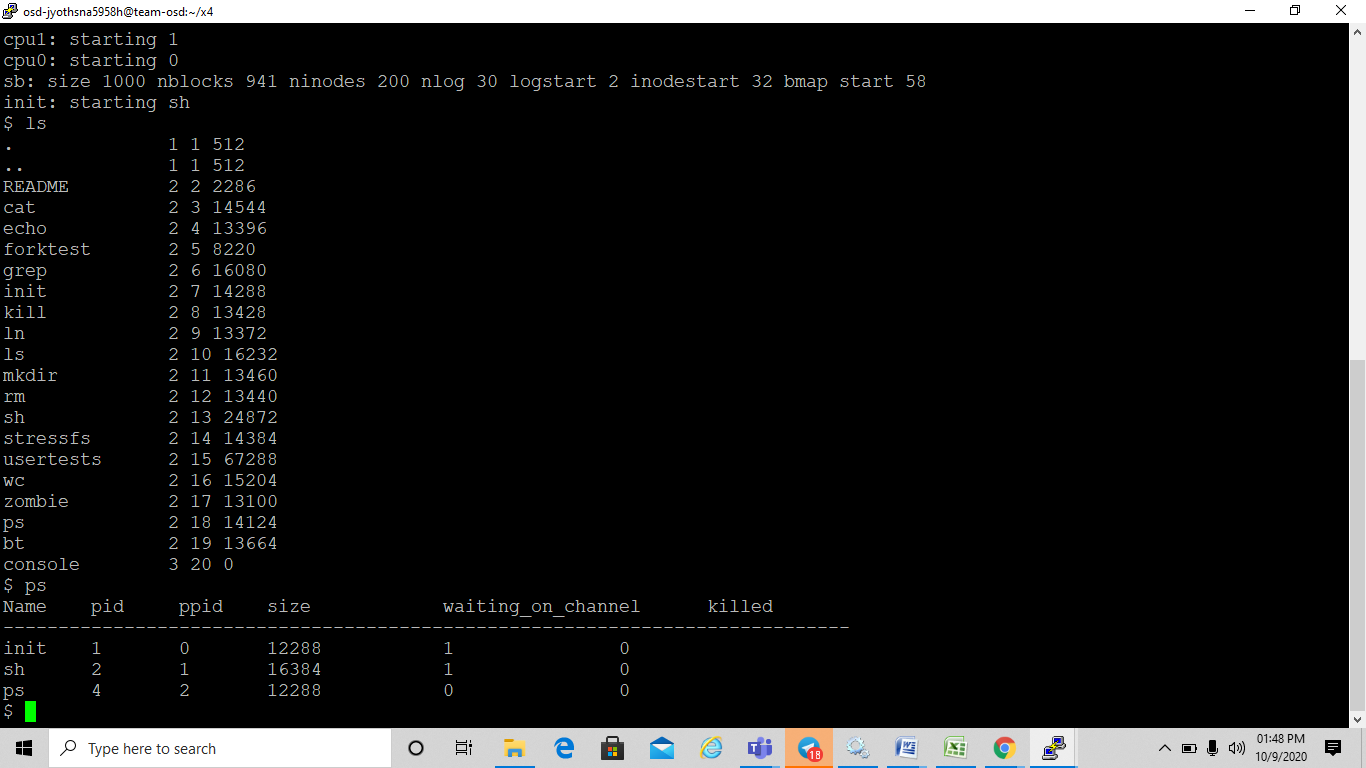
//In UPROGS\ section, Add

\_ps\

//In Extras Section, Add

ps.c

**Output:**

****