Operating Systems Project #1 - xv6 System Calls

SWE3004-42 Introduction to Operating Systems - Spring 2018

Due date: Mar 30 (Fri) 11:59pm

1 Goal

In this project, you will implement three system calls in xv6 - `getnice`, `setnice`, `ps`.

2 Synopsis

```
int getnice(int pid)
int setnice(int pid, int value)
void ps(int pid);
```

3 Description of the System Calls

- The `getnice` function obtains and returns the nice value of a process. It return -1 if there is no process corresponding to the pid.
- The `setnice` function sets the nice value of a process. The default nice value is 20. Lower nice values cause more favorable scheduling. The range of valid nice value is [0, 40]. If the nice value is invalid, do not change the nice value of the process. On success, `setnice` function returns 0. If there is no process corresponding to the pid of the nice value is invalid, it returns -1.
- The `ps` function prints out process(s)'s information, which includes pid, nice, status, and name of each process. If the pid is 0, print out all processes' information. Otherwise, print out corresponding process's information. If there is no process corresponding to the pid, print out nothing.

You must print out a process's information with the following `printf` format specifiers for ease of grading.

```
printf("%d %d %s %s\n", pid, niceness, state, process_name);
```

The state of a process can be `UNUSED`, `EMBRYO`, `SLEEPING`, `RUNNABLE`, `RUNNING`, or `ZOMBIE`.

4 How to Run xv6

4.1 Install Ubuntu

First of all, you need to download and install Ubuntu in your virtual machine and run the following commands to install compilers and git utilities. If you have any difficulty in setting up a Linux virtual machine in your PC or laptop, please let me know.

```
$ sudo apt-get upgrade
$ sudo apt-get install build-essential
$ sudo apt-get install gcc-multilib
$ sudo apt-get install git
```

4.2 Install QEMU

Xv6 should be installed along with QEMU virtual machine. So, please run the following commands in your Ubuntu.

```
$ git clone https://github.com/geofft/qemu.git -b 6.828-1.7.0
$ sudo apt-get install libsdl1.2-dev
$ sudo apt-get install autoconf
$ sudo apt-get install libtool
$ cd qemu
$ git submodule update --init pixman
$ ./configure --disable-kvm --target-list="i386-softmmu x86_64-softmmu" --disable-werror
$ make
$ sudo make install
```

5 How to Start Project

Download xv6 source code by running the following git command.

```
git clone https://github.com/jinsoox/xv6-skku.git -b pa1
```

In the xv6-skku directory, you will find a bunch of source code files. Among them, the followings are the files that you need to read and modify.

- **sysproc.c**: add the implementation of your method here
- **syscall.h**: define the position of the system call vector that connect to your implementation
- **user.h**: define the function that can be called through the shell
- **syscall.c**: external define the function that connect the shell and the kernel, use the position defined in syscall.h to add the function to the system call vector
- **usys.S**: use the macro to define connect the call of user to the system call function
- **defs.h**: add a forward declaration for your new system call
- **sysfunc.h**: add the new system call handler into this file too like `int sys_newsystemcall(void)`

Note that the xv6-skku is slightly different from the original xv6 source code in that it implemented a `halt` system call, and added a tarball target to Makefile so that it enables "make tarball" command.

To start xv6, you must run the following command.

```
$ cd xv6-public
$ make qemu-nox
```

Unfortunately, I haven't found out yet how to kill this shell once you started it. So, to kill xv6, you will have to run another shell in your Ubuntu and run `killall make` command to kill it.

6 How to Submit

To submit your project, you must run `make tarball` command in xv6-skku directory to compress your source codes into one .tar.gz file. This .tar.gz file must be uploaded to the iCampus assignment submission menu so that grading TAs can download it.

Note: Please don't forget writing your ID and project number in the Makefile before creating a tarball.

For any questions, please post them in iCampus or Piazza so that we can share your questions and answers with other students and TAs. As I mentioned in the first class, this is the first time that I teach OS and manage xv6 projects. So, there can be some mistakes and unclear requirements in the project. Please feel free to raise any issues and post any questions. Also, if you can answer other students' questions, you are welcome to post the answers. You would get some credits for posting questions and answering other students' questions.