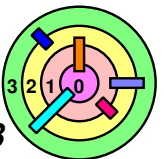


Kernel Programming Assignments (Part 1)

Bill Cheng

<http://merlot.usc.edu/william/usc/>

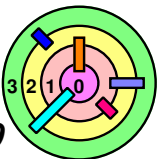


Academic Integrity Policy



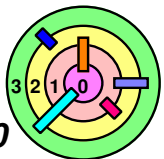
From the spec:

- ***accessing*** a submission from a previous semester is considered cheating
 - e.g., you run a submission from a previous semester to see how another student's code works
- if it's ***evident*** from your submission that you have ***accessed*** a submission from a previous semester, I will forward your case to USC Student Judicial Affairs for cheating investigation
- the standard punishment is a grade of **F** in the class.
- since there is only one submission from a team, ***even if only one team member cheated, the entire team is considered cheated.***
- therefore, please make sure that your teammates are not cheating!



Academic Integrity Policy

- ➡ Since the kernel assignments are group assignments, if *one of your teammate cheated*, the *entire group* is considered cheated
- ➡ it is *your responsibility* to make sure that *your teammates are not cheating*
 - ➡ if you noticed that your teammate has written some code he/she cannot explain, you need to delete the code and write from scratch!
 - ➡ if you know that your teammate has code from previous semester, it's best you ask your teammate to *throw away the code*
- ➡ As mentioned in Lecture 1, posting `weenix` code (whether it's the original code or your code or a combination) to *public* bitbucket (or something similar) is considered *cheating*
- ➡ posting warmup code to public bitbucket is also not permitted
 - ➡ you must *not* use `github.com`
 - ➡ can use `bitbucket.org`, but make sure it's private



Kernel Programming Assignments

- ➡ Tom Doeppner's *weenix* source and binary code
- provided as `weenix-assignment-3.8.0.tar.gz`
 - incomplete
 - contains code like:

```
NOT_YET_IMPLEMENTED("PROCS: bootstrap");
```

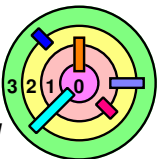
assignment name ————↑

function name —————↑

- your job is to implement these functions by *replacing* these lines with your code
 - ◆ please *replace* them *in-place*
 - ◆ do NOT *reorder* code there

- to look for such code:

```
grep PROCS: kernel/*.c
grep PROCS: kernel/*/*.c
grep PROCS: kernel/*/*/*.c
grep PROCS: kernel/*/*/*/*.c
```



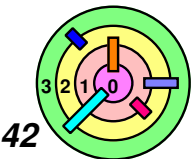
Download and Setup

```
% tar xvzf weenix-assignment-3.8.0.tar.gz
% cd weenix-assignment-3.8.0/weenix
% make clean
% make
% ./weenix -n
```

- ➡ Don't type the long commands, just copy from your web browser and paste into your terminal
- ▬ if something looks wrong, seek help as soon as you can
 - ▬ best *not* to do the above in a Shared Folder
 - see kernel FAQ if you have to do that

- ➡ Don't forget to install these when you setup Ubuntu 16.04:

```
sudo apt-get install -y git
sudo apt-get install -y gitk
sudo apt-get install -y git-core
sudo apt-get install -y qemu
sudo apt-get install -y cscope
```



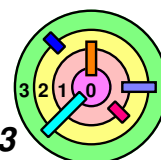
Download and Setup

- ➡ If all goes well, you should see tons of stuff fly by and the following at the bottom of the terminal:

```
Not yet implemented: PROCS: bootstrap, file
main/kmain.c, line 184
panic in main/kmain.c:186 bootstrap():
weenix returned to bootstrap()!!! BAD!!!
```

Kernel Halting.

- ➡ = press <Ctrl+C> in the terminal to quit
- ➡ Make sure you have tried the above this weekend
 - = any problem, let us know **NOW**
- ➡ Don't worry that you don't know how to do kernel 1
 - = for now, just get familiar with the spec and the documentation
 - and may be read some kernel code
 - you are **not** expected to understand a lot of the kernel code



Documentation

- ➡ The `weenix` documentation is in `doc/latex/documentation.pdf`
 - ▬ introduces `weenix` to you
 - ▬ detailed instructions on assignments
 - ▬ you must read it thoroughly
- ➡ We are doing three of the assignments
 - ▬ *Processes and Threads (PROCS)*
 - ▬ *Virtual File System (VFS)*
 - ▬ *Virtual Memory (VM)*
- ➡ We are *not* doing two of the assignments
 - ▬ Drivers (DRIVERS)
 - ▬ System V File System (S5FS)
 - ▬ these are done for you and they are compiled and provided as libraries
 - `kernel/libdrivers.a` and `kernel/libS5fs.a`
 - ◆ source code for these are *not* available; you need to learn how to *work around* code you don't have

