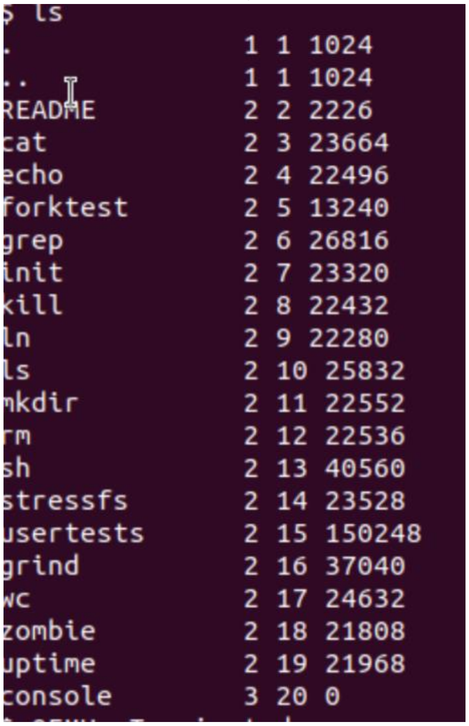
CS4375-13948 Fall 2023
Denisse Loya
dloyavilla@miners.utep.edu

Homework Report
Submitted on 18-Sep-23

HW 1: Introduction to xv6

Task 1. Boot xv6 and explore utilities

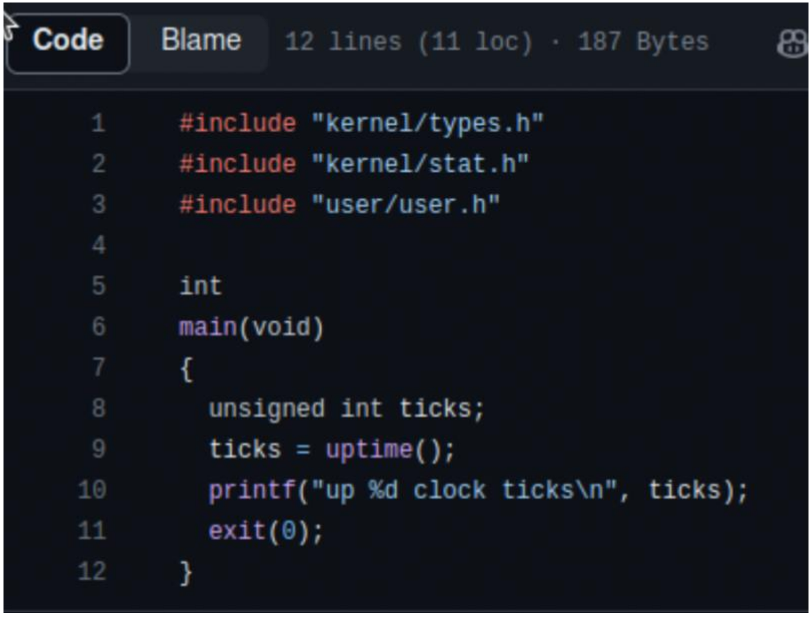
Mac with M1 2020, VMware Fusion Ubuntu 64 bit ARM 22.04.1



```
$ ls
.              1 1 1024
..             1 1 1024
README         2 2 2226
cat            2 3 23664
echo           2 4 22496
forktest       2 5 13240
grep           2 6 26816
init           2 7 23320
kill           2 8 22432
ln             2 9 22280
ls             2 10 25832
mkdir          2 11 22552
rm             2 12 22536
sh             2 13 40560
stressfs       2 14 23528
usertests      2 15 150248
grind          2 16 37040
wc             2 17 24632
zombie         2 18 21808
uptime         2 19 21968
console        3 20 0
```

Kill: this command interrupted or terminated the process
Echo: this command printed out the line that was passed as an argument
Mkdir: it created a new directory.

Task 2. Implement the uptime utility



```
#include "kernel/types.h"
#include "kernel/stat.h"
#include "user/user.h"

int
main(void)
{
  unsigned int ticks;
  ticks = uptime();
  printf("up %d clock ticks\n", ticks);
  exit(0);
}
```

What I learned?

I gained insights into the internals of an operating system xv6. I understand better how the kernel manages system information and how user-level programs can interact with the kernel. I learned about system calls and how they are used to bridge the gap between user-level programs and the kernel. In this case, to use the uptime system call, which involves understanding how to invoke it from a user-level program.

Difficulties and how did I overcame them

- My VMware lost network connection so I couldn't submit my work at first but with help of Dr Moor I could reconnect my network.
- GitHub I had little experience so I had to learn about mirroring repositories but this was easy after some tutorials.