CS4375-13948 Fall 2023 Report
Samuel Ramirez 9/22/23.

Introduction to xv6

My private xv6 repo: https://github.com/SamRamir/Myxv6

Task 1. Boot xv6 and explore utilities.

For my Linux VM setup I used Oracle VM Virtualbox and created a virtual machine using Ubuntu Linux.

My results of building and booting xv6 on RISC-V:

```
vboxuser@Ubuntu22:~/Myxv6.git/Myxv6$ make qemu
qemu-system-riscv64 -machine virt -bios none -kernel kernel/kernel -m 128M -smp 3 -nographic -drive file=fs.img,if=none,format=raw,i
d=x0 -device virtio-blk-device,drive=x0,bus=virtio-mmio-bus.0

xv6 kernel is booting

hart 2 starting
hart 1 starting
init: starting sh
$ ls
.              1 1 1024
..             1 1 1024
README         2 2 2226
cat            2 3 23720
echo           2 4 22560
forktest       2 5 13288
grep           2 6 26872
init           2 7 23360
kill           2 8 22480
ln             2 9 22352
ls             2 10 25904
mkdir          2 11 22616
rm             2 12 22600
sh             2 13 40624
stressfs       2 14 23576
usertests      2 15 150312
grind          2 16 37096
wc             2 17 24696
zombie         2 18 21856
sleep          2 19 22280
ps             2 20 23400
pstree         2 21 24360
pstest         2 22 23456
uptime         2 23 21984
console        3 24 0
$
```

Figure 1: Results of typing ls in xv6 running in emulated RISC-V

Three other commands I explored:

- **echo command –** this command when called prints the text into the console. After looking at the code, I can say that it's a program that takes some instructions you give it through the command line, puts them on the screen with spaces between them, and then moves to a new line when it's done.

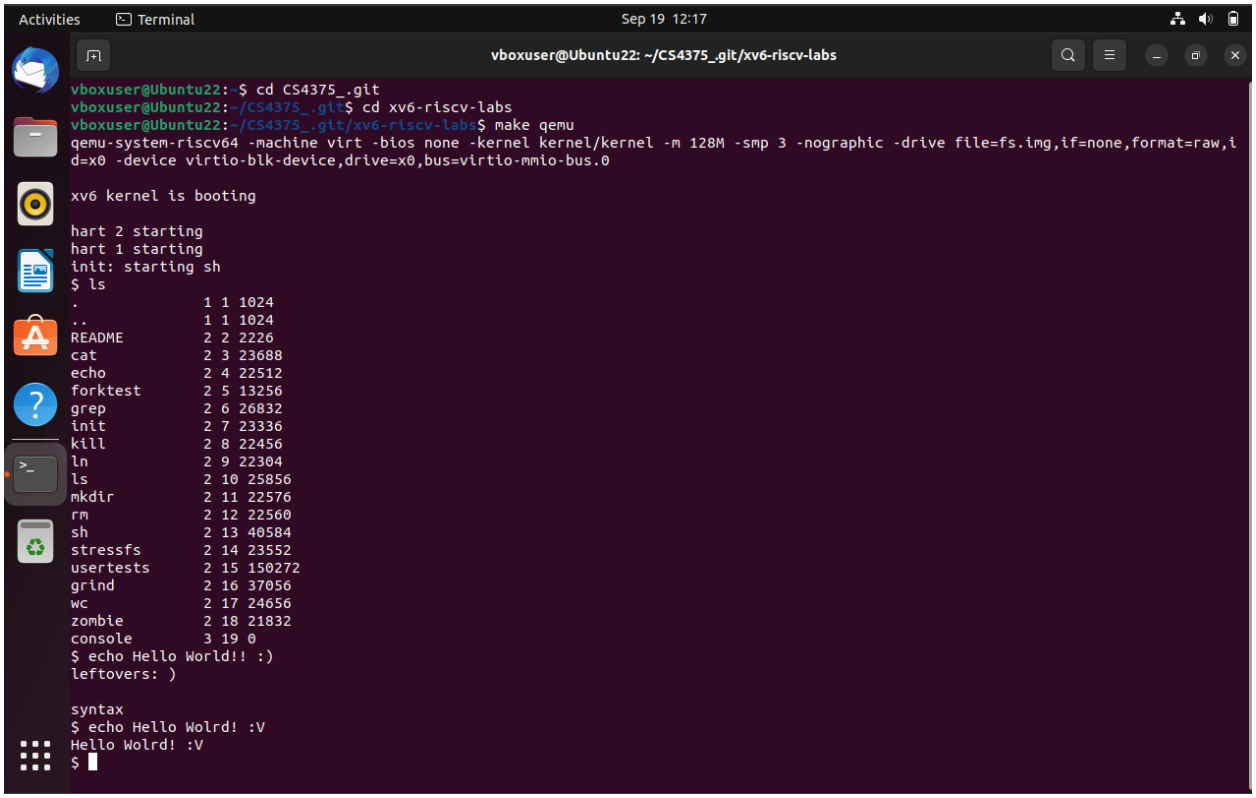

Figure 2: Results of echo command

- **cat command** – this command concatenates and displays the contents of the selected files. After looking and understanding the code the cat.c code can read and print the content that's inside the command- line argument. If there an error with the file it returns an error message or if you run it with no arguments, it will print what you type.

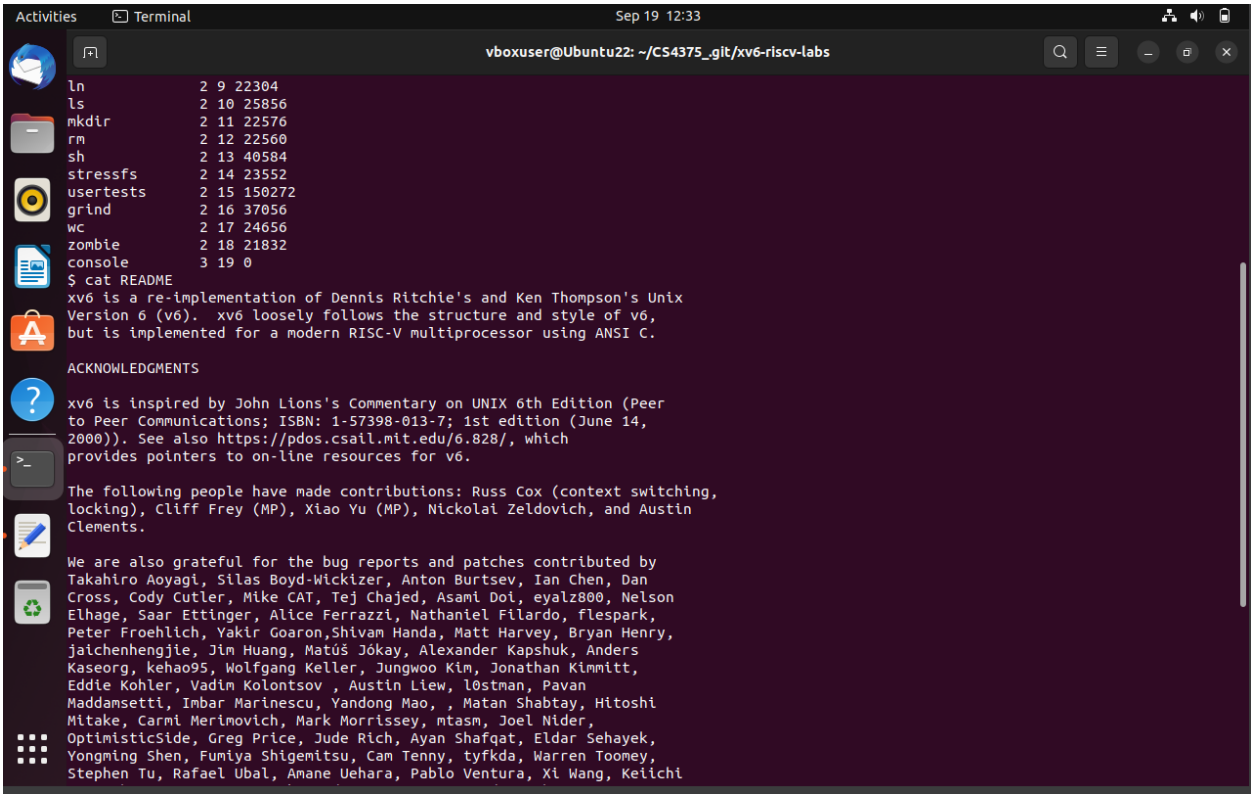

Figure 3: Results of cat command

- **mkdir command –** this command creates directories in the file system and names it to whatever you input when calling command. The mkdir code takes the input argument and creates a directory with that input. The code can give you an error if it cant create the directory. The code checks for the number of arguments and depending how many there are it can return an error or enter in a loop.

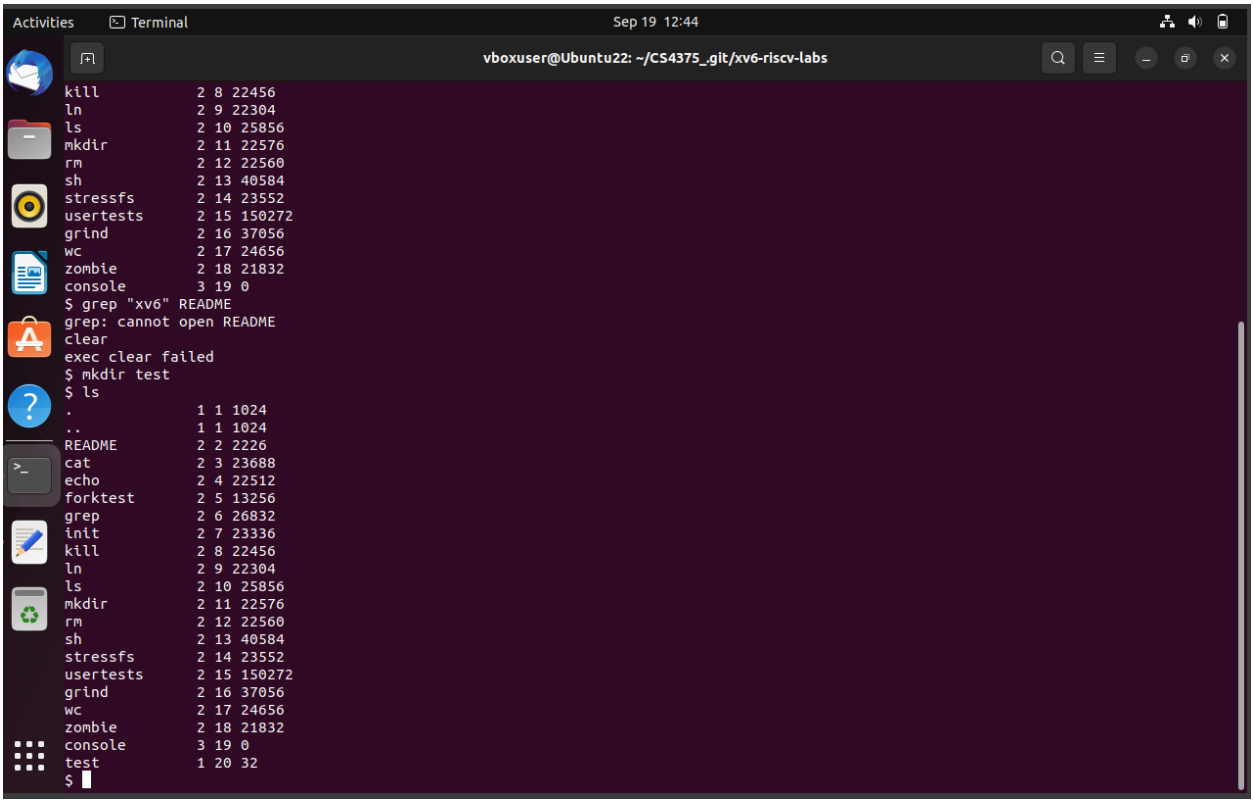

Figure 4: Results of mkdir command

Task 2. Implement the uptime utility

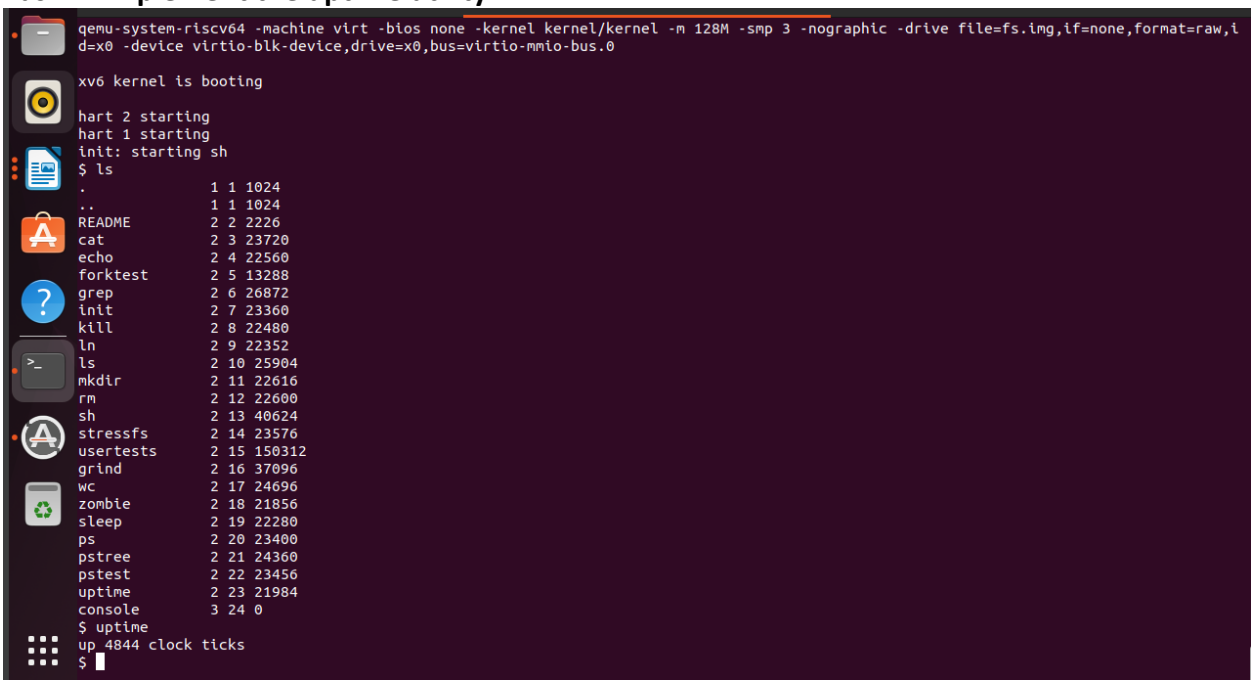

Figure 5: Results of uptime command

By doing this task, I figured out how to use commands in xv6 and how to make them. I also refreshed my memory on some of the Linux commands used in Ubuntu and how the C programming language works. I learned how to get into my project's repo and make my own separate branch for my code in the Linux terminal to keep things organized. I also learned how to make the 'uptime' command display the number of clock ticks since the system started.

Some of the difficulties I ran into was after creating the uptime.c file in in the user directory of xv6-riscv-labs and after adding it to the UPROGS. When running make qemu I would get error:

```
vboxuser@Ubuntu22:~/CS4375_.git/xv6-riscv-labs$ make qemu
riscv64-linux-gnu-gcc -Wall -Werror -O -fno-omit-frame-pointer -ggdb -MD -mcmode
l=medany -ffreestanding -fno-common -nostdlib -mno-relax -I. -fno-stack-protecto
r -fno-pie -no-pie   -c -o user/uptime.o user/uptime.c
In file included from user/uptime.c:1:
./kernel/stat.h:7:3: error: unknown type name ‘uint’
    7 |   uint ino;    // Inode number
      |   ^~~~
./kernel/stat.h:10:3: error: unknown type name ‘uint64’
   10 |   uint64 size; // Size of file in bytes
      |   ^~~~~~
make: *** [<builtin>: user/uptime.o] Error 1
```

Figure 6 : make qemu error after uptime

To fix this error I tried looking for any errors on my uptime.c file or in the Makefile but I couldn’t find what was wrong so at the end I had to remove uptime.c files and create them again to fix this error.

Another difficulty I ran into was cloning the repositories into the wrong directory but I was able to overcome it with github’s documentation.

Sources

https://docs.github.com/en/repositories/creating-and-managing-repositories/duplicating-a-repository

https://github.com/git-guides/git-push