CS153: Design of Operating Systems

You can use sledge server for the course labs.

$ ssh -X username@sledge.cs.ucr.edu

On MACOS:

$ ssh -Y username@sledge.cs.ucr.edu

Windows users may find "Mobaxterm" to be a good SSH and FTP client (the free version):

https://mobaxterm.mobatek.net/download.html

MAC users will need:

https://www.xquartz.org/

for GUI

To download xv6:

$ git clone https://github.com/mit-pdos/xv6-public.git

$ cd xv6-public

Note: whenever you want to discard all of your changes do this in xv6-public:

$ git checkout .

Links:

xv6 book
xv6 indexed/cross referenced code

To run XV6:

$ make qemu

If the mouse pointer gets stuck in the QEMU emulator window press: Ctrl + Alt + G

Hint: Always do "make clean" first
$ make clean && make qemu

You may find the QEMU window to be slow so it is recommended you do this instead:
$ make clean && make qemu-nox
To exit XV6, use “ctrl + a” then hit “x”

Note: you may find the text editor geany to be useful:

$ cd xv6-public/

$ geany *.c *.h *.S Makefile &

You can right click a function or variable to find where it has been defined and used in the code.

For a quick vim tutorial type in terminal:

$ vimtutor

If you want to use vim instead of geany, use ctags to help you index the code:

https://andrew.stwrt.ca/posts/vim-ctags/

This grep command is helpful:

$ grep -rnw –color . -e “search string”

https://www.cyberciti.biz/faq/howto-use-grep-command-in-linux-unix/

hint: do “make clean” before running grep and be aware that grep results will conflict with the ctags generated file.

To run Qemu with GDB

You need to do this the first time only, in your home dir:

$ cd ~

$ echo "add-auto-load-safe-path /home/csmajs/NetID/xv6-public/.gdbinit" >> .gdbinit

I am assuming that your xv6 folder is placed in your home directory and named “xv6-public”

Otherwise here is a generic one;

$ cd ~

$ echo "add-auto-load-safe-path /home/csmajs/NetID/PATH_TO_XV6_FOLDER/.gdbinit" >> .gdbinit

After that you need to open another terminal at the same xv6-public folder:

$ xfce4-terminal &

In the original terminal type:

$ cd xv6-public

$ make qemu-gdb

Or $ make qemu-nox-gdb

In the second terminal type:

$ cd xv6-public
$ gdb

+ target remote localhost:25049
The target architecture is assumed to be i8086 [f000:fff0] 0xffff0: ljmp $0xf000,$0xe05b
0x0000fff0 in ?? ()
+ symbol-file kernel
$ (gdb) continue

For GDB with GUI, type:

$ gdb -tui

Here is a GBD tutorial:

http://www.gdbtutorial.com