The Bochs Environment with GDB Integration:

1. Download Bochs with gdb source code from the following location:
    a. I downloaded the bochs-2.6.8.tar.gz from the website

2. Configure Bochs with gdb stub enabled, under the directory of the Bochs source code:
    a. sudo sh .conf.linux --enable-gdb-stub

3. After the configuration, sudo make - throws 2 errors
    a. There are a couple of missing libraries,
        i. Sudo apt-get install libx11-dev
        ii. Sudo apt-get install librandr-dev

    b. uncommenting which config=normal and commenting out which config=plugins in .conf.linux.

4. enable the gdb stub in the Bochs configuration file
    a. gdbstub: enabled=1, port=1234, text_base=0, data_base=0, bss_base=0

5. ELF (Executable and Linkable Format) output that retains the debugging information.
    a. Remove the first line of the linker.ld file (OUTPUT_FORMAT(“binary”)

    b. In makefile add -g flag:
        i. $(GCC) $(GCC_OPTIONS) -g -c -o utils.o utils.C
        ii. $(GCC) $(GCC_OPTIONS) -g -c -o console.o console.C
        iii. $(GCC) $(GCC_OPTIONS) -g -c -o kernel.o kernel.C

    c. In makefile and copykernel files make appropriate changes i.e, rename kernel.bin to Kernel.elf

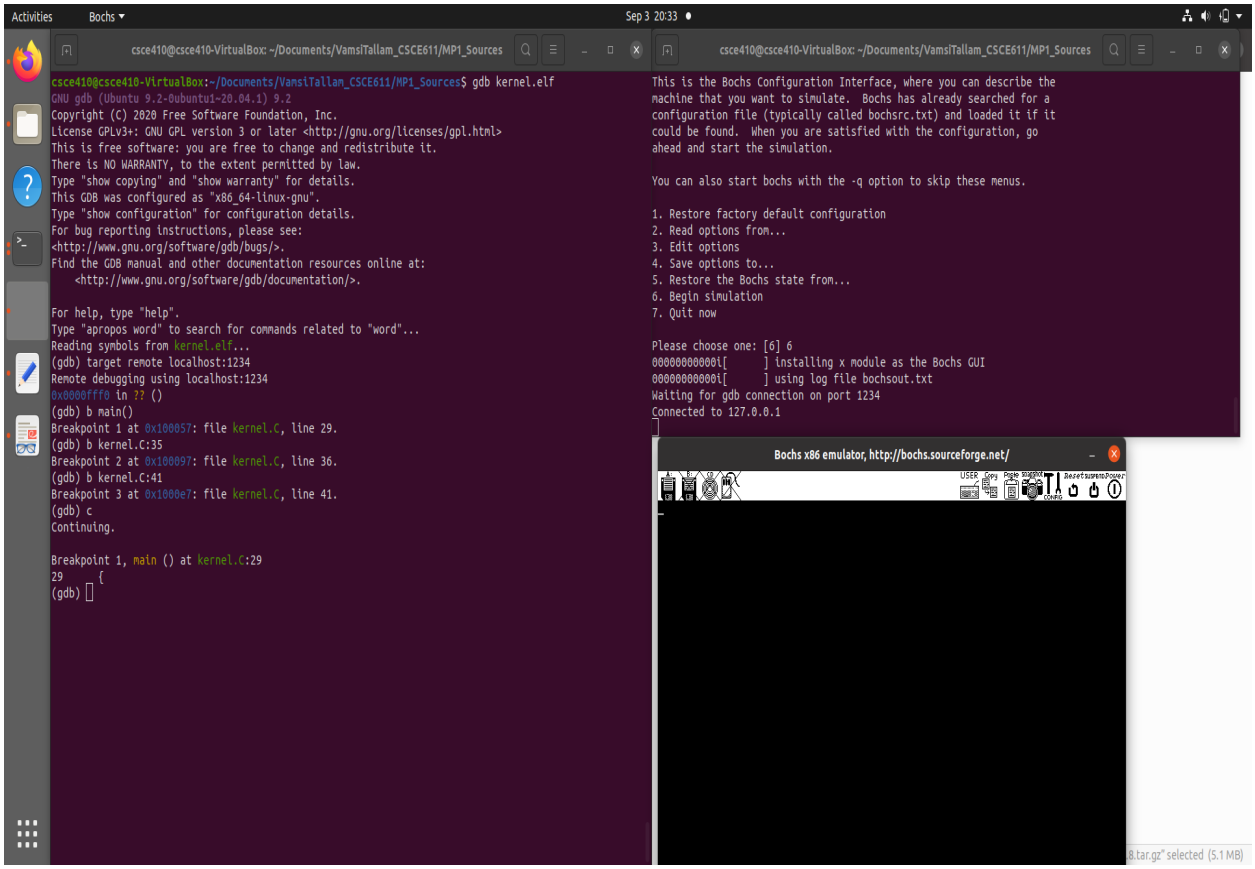
    d. Make clean

    e. Make

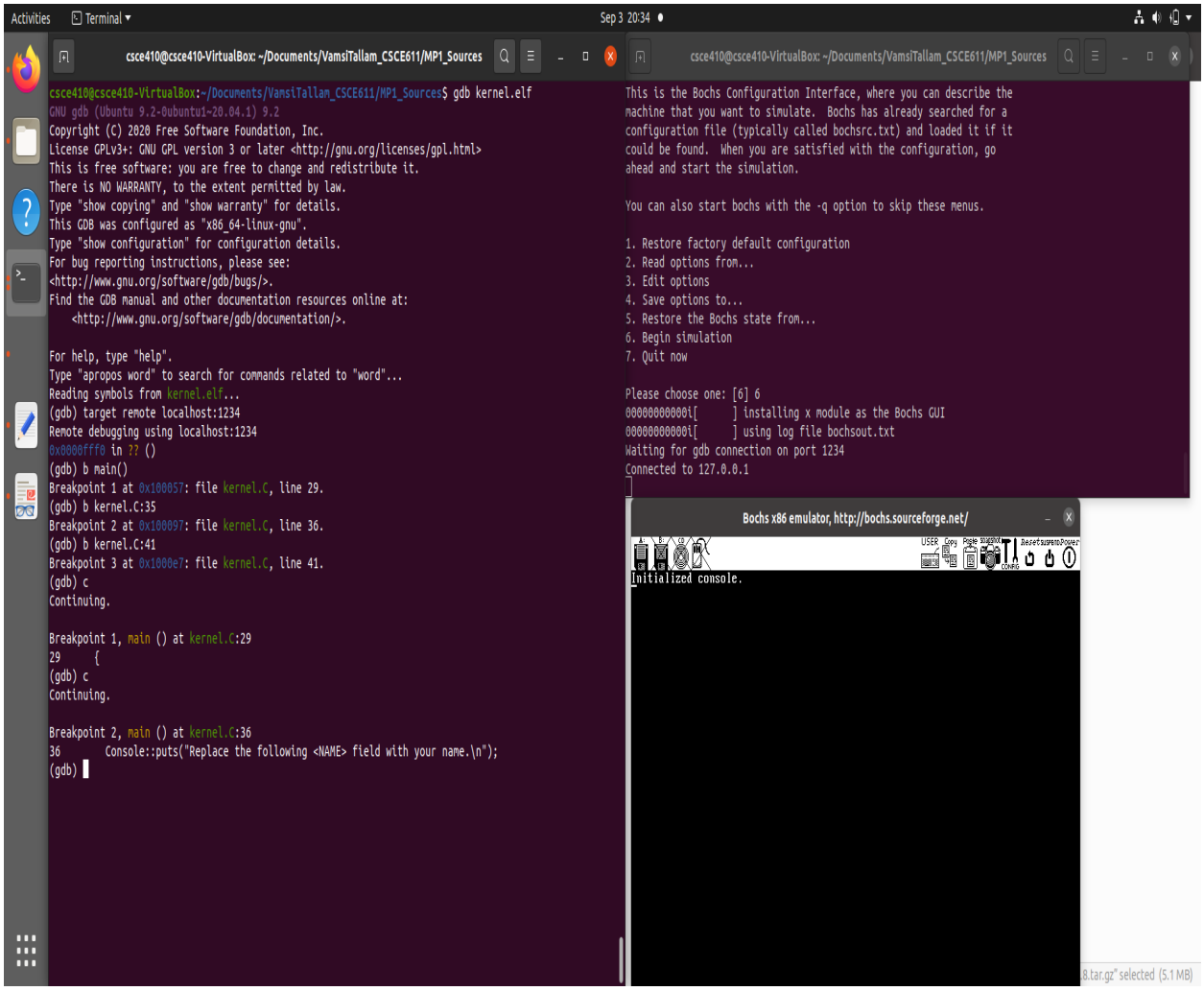
    f. ./copykernel.sh

6. To load up Bochs
    a. bochs -f bochsrc.bxrc - select option 6

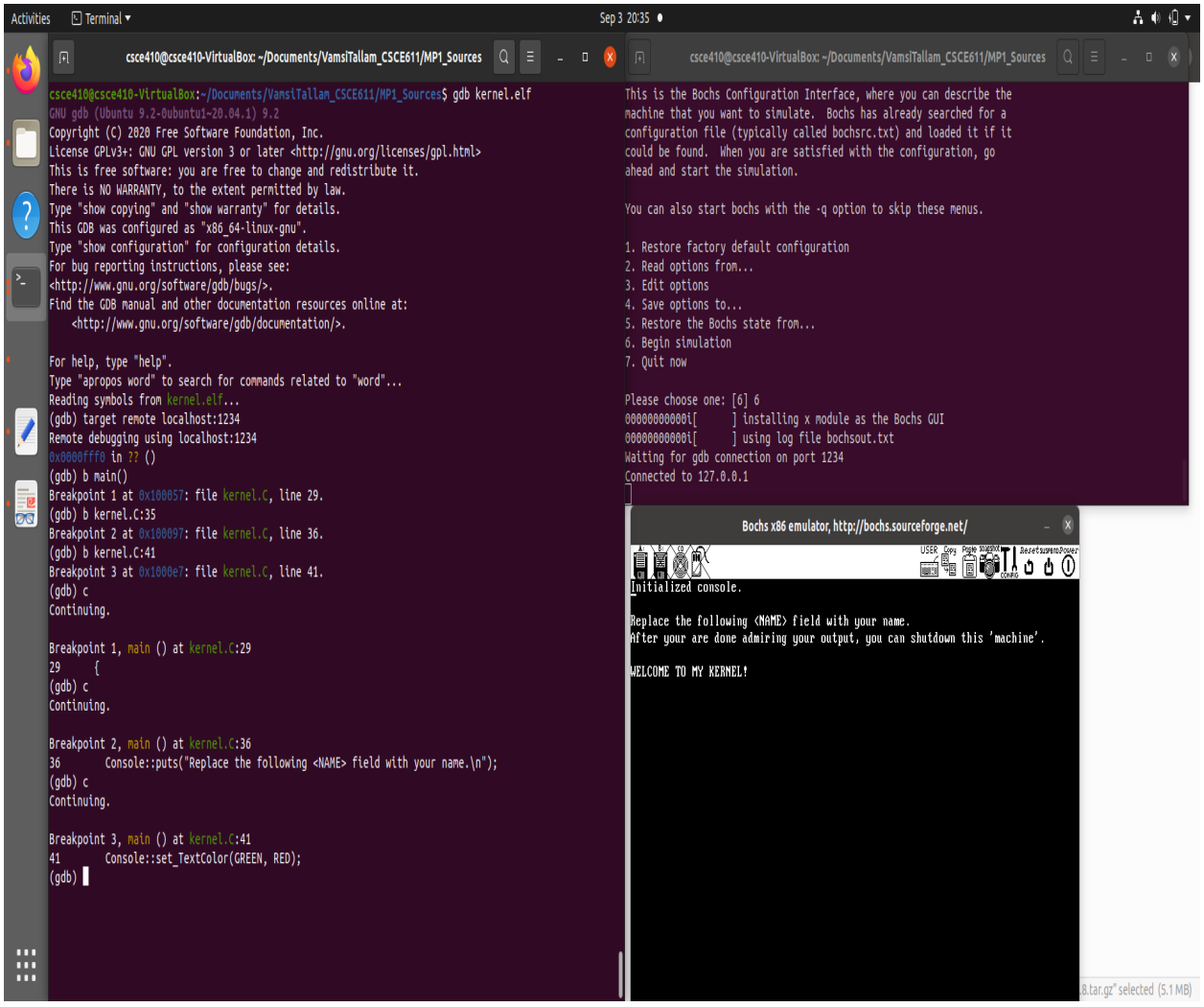
b. To connect from GDB, open a new terminal and run
   i. Gdb kernel.elf
   ii. (gdb) target remote localhost:1234
   iii. (gdb) b main()
   iv. (gdb) b kernel.C:35
   v. (gdb) b kernel.C:41
   vi. C
   vii. q



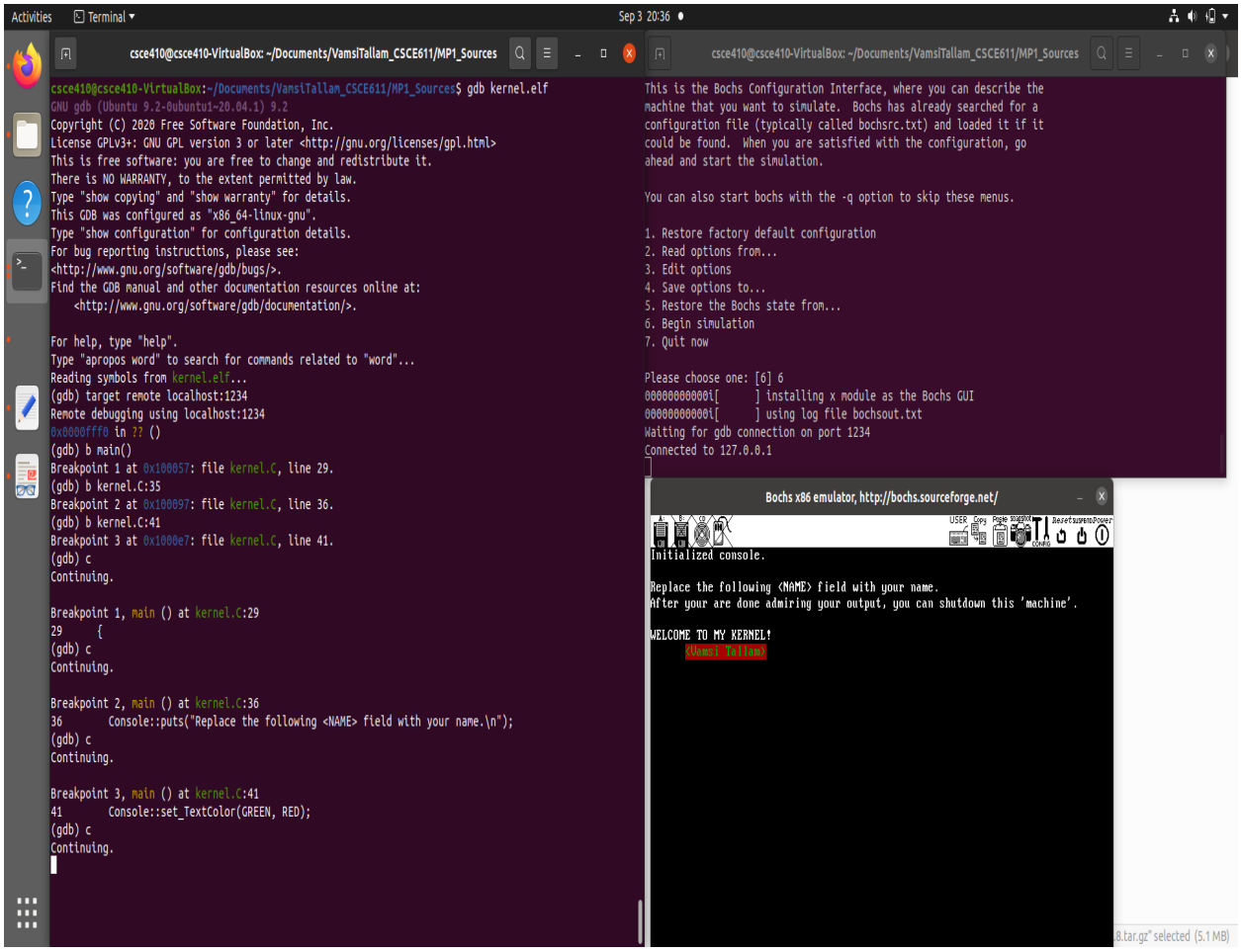
Screenshot capturing the run till first breakpoint



Screenshot capturing run till second breakpoint



Screenshot capturing run till third breakpoint



Screenshot capturing the run from third breakpoint to the end of code.