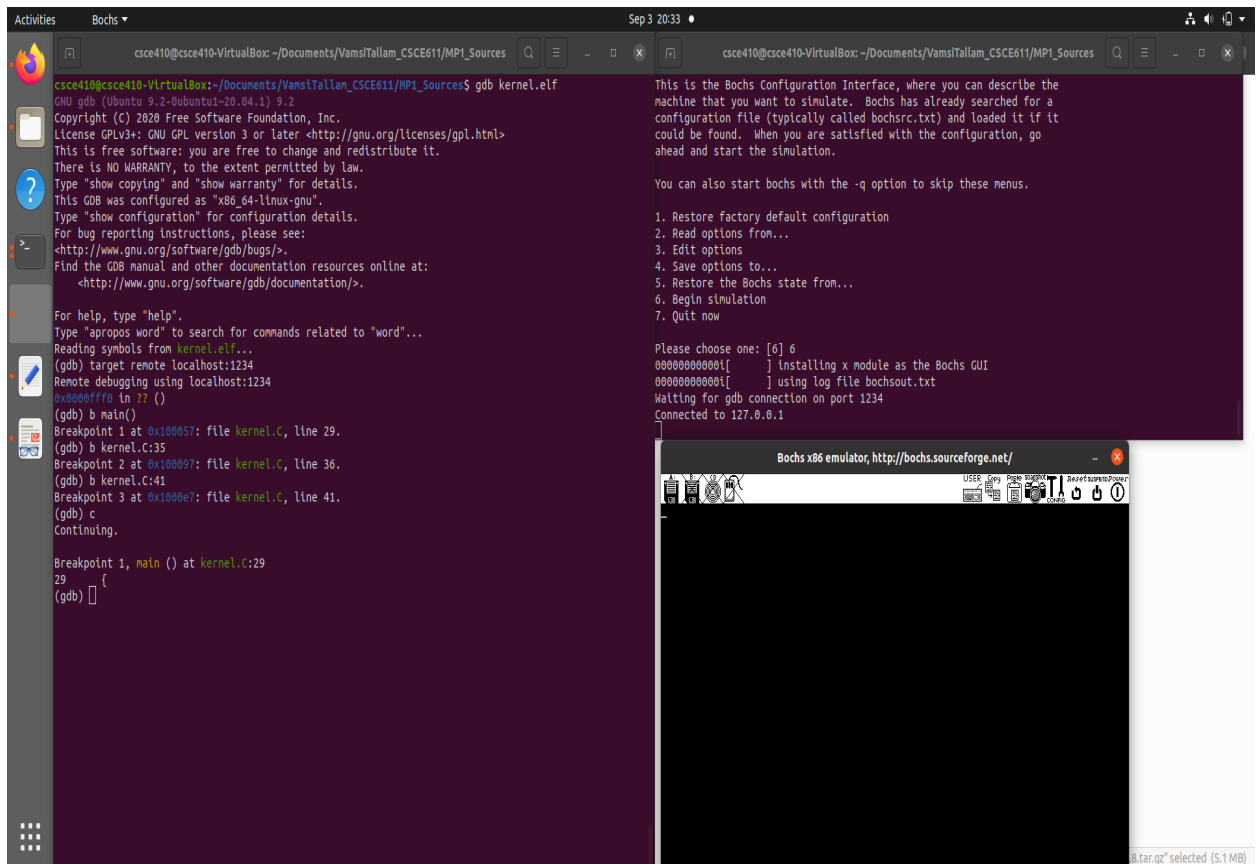


## The Bochs Environment with GDB Integration:

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
- Gdb kernel.elf
  - (gdb) target remote localhost:1234
  - (gdb) b main()
  - (gdb) b kernel.C:35
  - (gdb) b kernel.C:41
  - C
  - q



```
csce410@csce410-VirtualBox: ~/Documents/VamsiTallam_CSCE611/MP1_Sources
GNU gdb (Ubuntu 9.2-0ubuntu1-20.04.1) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software; you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from kernel.elf...
(gdb) target remote localhost:1234
Remote debugging using localhost:1234
0x0000ffff in ?? ()
(gdb) b main()
Breakpoint 1 at 0x100057: file kernel.C, line 29.
(gdb) b kernel.C:35
Breakpoint 2 at 0x100097: file kernel.C, line 36.
(gdb) b kernel.C:41
Breakpoint 3 at 0x1000e7: file kernel.C, line 41.
(gdb) c
Continuing.

Breakpoint 1, main () at kernel.C:29
29 {
(gdb)

This is the Bochs Configuration Interface, where you can describe the
machine that you want to simulate.  Bochs has already searched for a
configuration file (typically called bochsrc.txt) and loaded it if it
could be found.  When you are satisfied with the configuration, go
ahead and start the simulation.

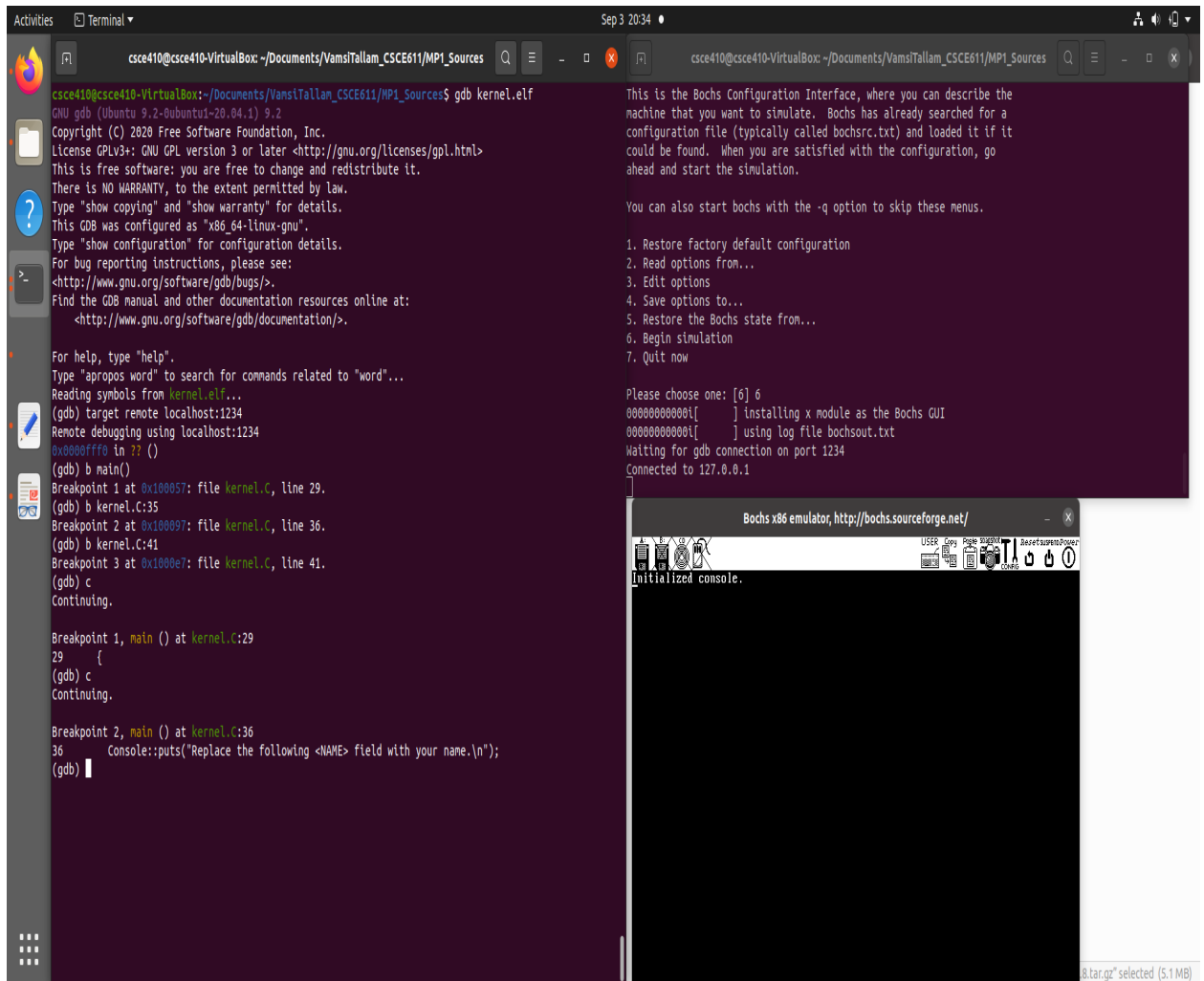
You can also start bochs with the -q option to skip these menus.

1. Restore factory default configuration
2. Read options from...
3. Edit options
4. Save options to...
5. Restore the Bochs state from...
6. Begin simulation
7. Quit now

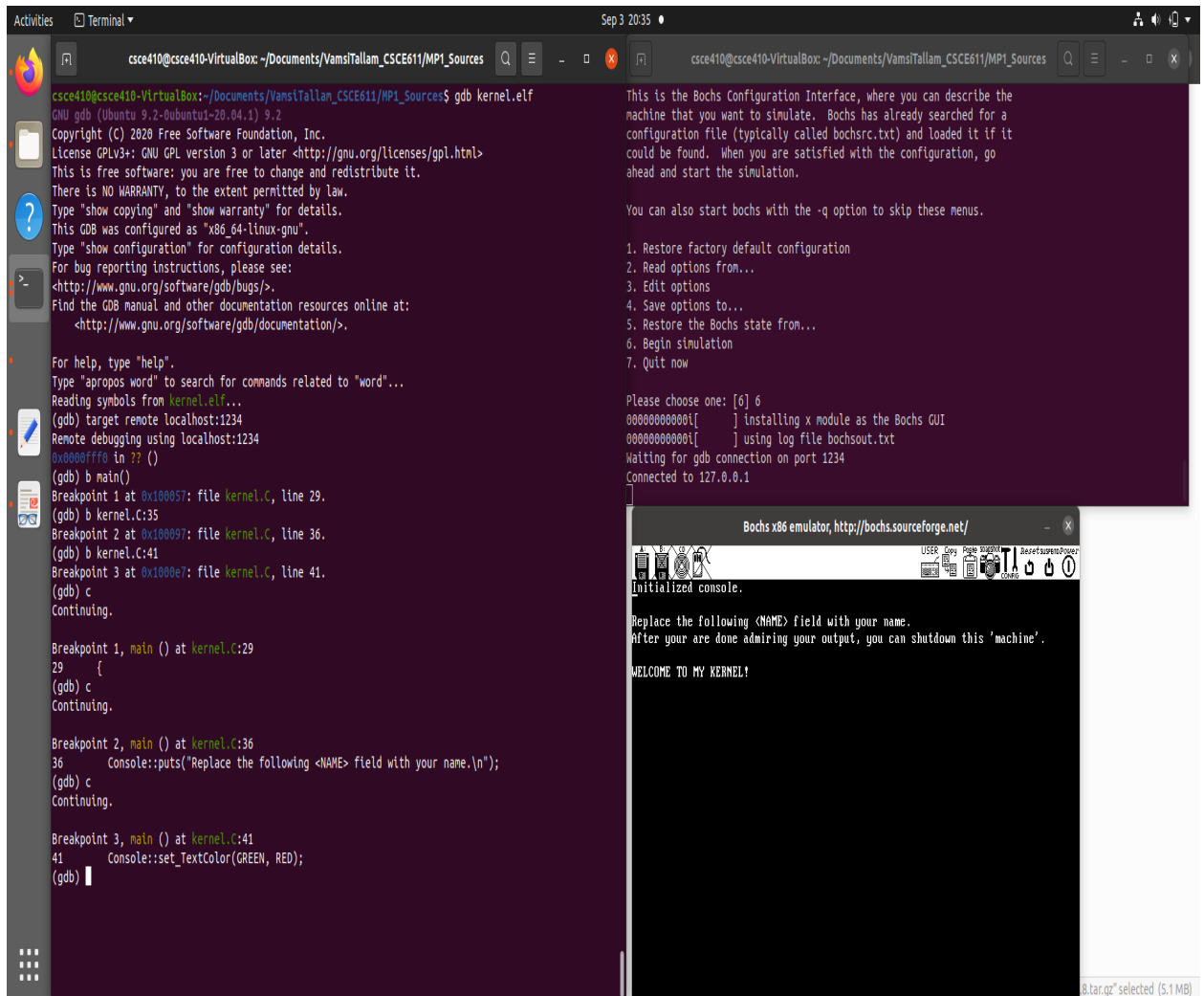
Please choose one: [6] 6
000000000000[ ] installing x module as the Bochs GUI
000000000000[ ] using log file bochsout.txt
Waiting for gdb connection on port 1234
Connected to 127.0.0.1

Bochs x86 emulator, http://bochs.sourceforge.net/
USER: CPU: PNP: RESET: POWER:
3 target selected (5.1 MB)
```

Screenshot capturing the run till first breakpoint



Screenshot capturing run till second breakpoint



Screenshot capturing run till third breakpoint

