For this exercise gdb was used to run dynamic analysis on the program. First, I displayed the disassembled code in gdb. Since it was unreadable, the disassembly-flavour was set to intel. At that point the disassembled code looked like the assembly displayed in binary ninja. Next a breakpoint was set at main. The program was then run, where it immediately stopped at the breakpoint. I displayed the disassembled code, which showed the functions being called. In that disassembled code, I noticed there was checkPass function. Using si, I went instruction by instruction until I reached the function call to scanf. For that I used ni, and it asked me to enter a passcode. I entered "asd". Next I kept using si until I entered the checkPass function. Inside that function, the disassembled code was displayed. In that strcmp was being called and within rsi data was being loaded. I then used si, until I got to the instruction after rsi was loaded with data. Following this, info registers was used to confirm the data was stored in rsi. Then the instruction x/8s $rsi was used to display the data stored in rsi as string. Here the passcode "xXShockwaveNSXx" was found. Following this, the program was ran as normal and the passcode was entered. From this the flag flag{Are\_You\_Doing\_COMP9417\_Next\_Term?} was obtained.





