CSE231 - Operating Systems

Assignment-5.1

Name: Ananya Lohani
Roll Number: 2019018
Branch: CSE

Description and Logic:

I have implemented my own bootloader for QEMU. The code sets up the VGA text mode(VGA buffer is used to store text), loads the global descriptor table, sets up the protected mode bit in register cr0, and then jumps to the code segment to print the output. I have initialized the global descriptor table and its code and data segments. Then we shift to the protected mode. The message is stored in register esi, and the VGA buffer is set to point to address 0xb8000. Then we loop over all characters in the message and print it on the screen. After there are no more characters left, we jump to stop and print the value stored in register cr0. Then finally we clear the interrupt flag and halt execution of the program.
Note that the value of register cr0 will be output as the ASCII value and not the numerical value.

Makefile:

The bootloader is assembled using the following command:

```
nasm boot.asm -f bin -o boot.bin
```

The bootloader is booted into QEMU using the following command:

```
qemu-system-i386 -fda boot.bin
```

Booting Instructions:

To make the code bootable, we use the following instructions:

```
times 510 - ($ - $$) db 0       ; pad the remaining bytes with 0
dw 0xaa55                       ; Magic number for loading the 512 byte sector
                                ; as a bootable image
```