

Write Up

Q1: As given in the question we have to go to protected mode (32 bit) and print hello world and value of CR0 register to do so initially we start in real mode (i.e 16 bit). Then we make some changes and move to 32 bit mode.

For this we first enable the A20 line so that we can access more than 1 mb of memory. Now we enable 32 bit instruction and use a 32 bit register by entering protected mode.

Now before this we will set up a Global descriptor table and we load it using lgdt instruction.

Global descriptor table tells us about the CPU memory segment and using lgdt instruction we can get a pointer to that table.

Next we set up gdt and then we print our string and value of the CR0 register.

Q2:

For this question as we just have to implement a normal text editor i used normal file handling in this . U can edit and read a file in this text editor. In editing mode you can append text to current text.

Whenever someone tries to read or write an already open file then it shows a warning.

To implement opening and writing to files I used fopen, fclose, fgetc, fputc function of C library. Then to lock the i used flock and LOCK_EX(EX for exclusive), LOCK_NB(NB for non-blocking), LOCK_UN(UN for unlock), present in the C library.

After all this there is one problem with code whenever i press CTRL+S it got stuck after going through some i got that when we press CTRL+S in terminal it stops taking input. So to disable this feature there is a function called enableRawMode. This disabled the original function of CTRL+S now when i press CTRL+S it gives its ASCII code i.e 19.

I also created a shell script, it will open 2 terminal windows and run makefile given.

Acknowledgements :

<https://viewsourcecode.org/snaptoken/kilo/>