1 DESIGN

Our solution first checks that the list is not empty. If it is, it fetches the next I/O request from the queue. If the list is empty, add the item anywhere in the queue. Otherwise, generate a list request to iterate. If the request sector position is greater than current position, add the item to the list directly before the current request position via insertion sort.

2 QUESTIONS

1) **What do you think the main point of this assignment is?** The main point of this assignment was to enhance our understanding of the CPU scheduler. It also helped familiarize is with the VM and how the pieces come together.
2) **How did you personally approach the problem? Design decisions, algorithm, etc.** We modified the noop-iosched.c file to make it a shortest seek time first implementation. The scheduler needed to be aware of the positioning at all times since the behaviour is largely determined by the sector position relative to the current position.
3) **How did you ensure your solution was correct? Testing details, for instance.** We ensured our solution was correct by writing into a testfile. We changed the scheduler using the make menuconfig command. To ensure we had changed the scheduler, we echo'd statements into that file and read out the contents to the terminal to see if the contents matched the expected output. We also implemented print statements into our sstf-iosched.c file to track what is happening to the list and which sector number is dispatched.
4) **What did you learn?** We learned about the VM and how the VM chooses which scheduler to use. We learned about the basic structure of a scheduler and how it merges and sorts. We made a new scheduler available to the VM as well and figured out how to compile the scheduler. We also learned various facts in the process such as the advantages of different scheduling algorithms.

3 VERSION CONTROL

3.0.1 Version Control (Link:https://github.com/avisinha1/cs444/commits/master)

Who	Work	Date
Omar	noop-iosched.c uploaded	10/30/2017
Omar	sstf-iosched.c uploaded. Almost complete but needs comments	10/30/2017
Omar	Finished sstf-iosched.c. Added comments	10/30/2017
Omar	Added Kconfig.iosched	10/30/2017
Omar	Added patch	10/30/2017
Omar	added tex and makefile	10/30/2017

4 WORK LOG

October 25th 2017: Began reading assignment details

October 26th 2017: Understood what the problem was asking and started a development of a solution based on noop

October 27th 2017: began implementation of LOOK algorithms

October 28th 2017: Started documentation of our work so it could be added to the write up

October 28th 2017: Began figuring out how to run the test and properly implement patch

October 29th 2017: Started putting content into the latex file so it would be complete by the due date

October 29th 2017: continued work on the patch and so it be prepared for testing

October 30th 2017: Began testing of patch and getting the results

October 30th 2017: Added data from version control log into the LaTex as well as completed linux patch

October 30th 2017: Finished LaTex write up and turned in on TEACH as a tarball