/*

****

Name : Nishit Sadhwani

UFID : 5393-5598

Email: nishit@cise.ufl.edu , n.sadhwani@ufl.edu

Subject Code : COP 4600

Project 3 :

To implement sequential files, and perform normal operations on it like (create, open ,delete, read and write).

This document explains, how I implemented the project.

This document also describes how testing and debugging was done and the errors.

****

*/

Programming :

Step 1 :I basically worked on making the application , which tests all the functions, like create , read , unlink , write , display

etc.

Step 2 : I then added flags for my file I_SEQ at various places, I then studied the way a regular file works, then I added the similar

flags for my sequential file.

Step 3 : After I was able to atleast create and open the sequential file , I inserted the patch provided by TA.

Step 4 : After the patch complied, I started working on read_map.

Step 5 : I figured out how read_map is going to work, I made the changes accordingly for the sequential file , it's basically that

you need to lookup in the INODE table for the first zone, and then you need to see the first block of every zone, figure out

the next zone and then traverse the list to find the first block.

Step 6 : After doing the read map, I switched over to new_block for writing, I then figured out an algorithm to allocate a new zone.

Step 7 : After allocating the new zone everytime it needed it, I moved over to truncate_inode, I then made changes to truncate_inode.

Step 8 :I then added flags for FNCTL, I used 4 different flags for different operations.

Step 9 : I have attached a file indicating all the changes.

Testing and Debugging :

I used various printf statements to see how things are working, like printing the directory in which the code is.

That helped me to understand the flow structure. I also used printf statements to print the value of different

variables like so that I could know the return values and which functions are working or not.

I also used printf statements at various modules to see if correct values are being returned.

Errors and bugs :

I had to deal with plethora of errors, some of them which I remember are :

- No space on device 3/176 error

  This error was probably because I used up all the space allocated to /root

- The write function didn't work, it was not writing any bytes to it.

-Memory Core Dumped :

This was probably because I didn't use a proper address or didn't use an address at all.

Output :

I have attached an output file as well, which shows the successful implementation of all the aspects of project 3.