What is not implemented in a3 are ext2_rm and ext2_mkdir, the other two - ext2_ln and ext2_cp are only partially implemented. For ext2_rm and ext2_mkdir I was not able to get to implement them but i wrote the pseudo code that would have helped me to implement them, just to show that i did think about how to solve them but unfortunately ext2_cp took me way too long to implement and i run out of time. I was taking six course and underestimated how much work was to implement a partial file system.

For ext2_cp and ext2_ln i wrote partially some runnable code that does part of the functionality asked for, but as the deadline approached i had to write more and more pseudo code. I spended most of my time trying to organize the helper functions and make them general enough so that i could use them in all functions. But as i got to the last modifications to modify the bitmaps everything went terribly wrong, i scrambled both my inode bitmap and my data bitmap so i was only able to make them functional. After that i was fona change the superblock as the last component to fully implement ext2_cp, but without the proper bitmaps i was unable to. So i decided move on to Ext2_ln i implemented as much as i could, after realizing that i was not going to finish i decided to do the pseudo code for the remaining functions.

Features completed:
Ext2_cy

- strip the paths and traverse through the file system
- finding free free inodes
- free data blocks
- **create a new directory entry**
- get data from the inodes (i.e their data blocks)
- add the entry to the image
- **create new inodes**
- **write my new directoryentry to the img**

Ext2_ln

- able to read directory entries
- traverse through the source and destination paths
- gather data to create a new directory entry connected to the destinations inode.
- rest is in pseudo code

used to grace days.