Kumaravinash Konduru

Kxk151330

**Project 3 - Summary**

When it came to doing this project I felt that I was able to learn a whole lot about how file allocation works. And though I also learned about most of these methods from the lectures and the slides provided, actually having to implement these methods into an actual project I felt allowed me to a greater extent learn and understand that material. That being said that things that I was mainly able to learn from doing this project included how to implement various kinds of file allocation methods. These included the contiguous file allocation method, the chained file allocation methods, and the indexed file allocation methods.

To allow for a file to be stored in the disk through a contiguous allocation methods I had to create a way to split up the file and then have each of the blocks be in a contiguous ordering within the disk. Thus the way by which I was able to implement a contiguous file allocation methods was by once the file was split I would then check the disk to see if there was a space large enough to fit the amount of blocks that the file would take up. Once that space was found within the disk the program would then take the first part of the file and then put it into the starting block of the sequence Then sequentially all of the rest of the file would then the put into the next blocks within the sequence until all of the file was put inside a contiguous set of blocks within the disk. Because of the nature of how the contiguous file allocation worked and because of the fact that it was the first file allocation method that I worked on in my program it made it so that I felt that it was the hard file allocation method for me to implement. I felt that this was the case due to the fact that for this file allocation method I had to create a method and an algorithm that searches the disk to see if there was a viable spot to put my file depending main on how many block would be needed to store the file. Initially I thought that I might just be able to implement a brute force solution to this problem where in I would just construct an algorithm that would search to see if each individual spot necessary would be available in order, but this algorithm seemed to take too long and was very hard to program and get right, so eventually I instead went with a more thoughtful approach, that being where I instead used an algorithm similar to one used for the maximum subarray problem. Using an algorithm based on the maximum subarray problem I was able to realized and construct a much more simple, efficient, and elegant solution to the problem that I was have. After I was able to figure out how to place the file portions into the appropriate blocks I then had a little bit of through trying to understand how I would be able to both collect the blocks together again in the correct order as to have the ability to display them again and also have the ability to delete the file by again collecting the blocks in order and then deleting or wiping the contents of the respective block. Initially I thought of doing the collection part of these parts by storing these blocks as a node but I ultimately decided to go with using a simple array that would loop though the necessary blocks by using the information stored within the file table and the bitmap.

After I was able to finish the hurdle that was the contiguous file allocation method I then had to complete both the chained file allocation method, which is a file allocation method where the file is split into separate sections and the sections are placed into block of the disk randomly and have some connection that leads them from one block to the next in the sequence, and the indexed file allocation method, which is a file allocation method that means that the file is split into an equally proportioned section which are then randomly placed into blocks of the disk with an index block acting as a guide to showing in which blocks the seperate piece of the file are and within what order those piece of the file are. Ultimately though once I was able to write and complete all of the necessary methods for the contiguous file allocation method, both the chained file allocation method and the indexed file allocation method came much more easily to me.