The search tool works by constantly checking and writing to cache. When a word/list of words is given to search our program will first check the cache which is a linkedlist of word nodes in which each node contains a linkedlist of file nodes. The word node keeps track of the amount of memory the linked list of file nodes takes up. We will first check if the word exists in cache by calling checkCache, if it does we store the file results and go to the next word. Otherwise our program will call getFileList which will respond with a list of files where that word is found, and that node is added to cache. Before the word is added to cache we make sure the cache isn't exceding it's memory limit, if the memory is full we follow the FIFO method of clearing memory in the cache and subtract from the global current cache size variable. Once the word is added we update the global variable of current cache size. At the end we, would free the memory, but it's segfaulting when we try to do so. As such, there are memory leaks.