Rakshith Raghu, rr5de, 10/23/2018

My implementation was faulty coming to the inlab. This is because I incorrectly declared the dictionary vector in the hashtable declaration. This caused the type element dictionary not to be implemented. I fixed this in lab, and am wondering if I should ask for a regrade of the prelab, resubmitting this file. There does appear to be an issue with repeating the word it finds. It probably something to do with the last for loop regarding length of string pulled from grid that inflates the number of found words. The issue is that near the end of the grid, it does discriminate the distance. So at a corner, length 3 and length 10 would produce the same string. This inflates the number of words found slightly. This happens sporadically, and will be dealt with during the postlab.

Being blunt about it, my program took a huge time to run. The main issue, on top of the 4 for loops (and the 5th for loop) was when it compared to the dictionary, it effectively checked against every element of the dictionary. The reason I did this was because I chose to do linear collision, which means an element of the dictionary could be stored at a very different place than the initial key. I believe I can optimize this by switching to buckets rather than linear collision, which allow me to just grab an initial key, and then search against every element of that bucket rather than the whole list.

the big O run time would be: O(rows * columns * directions * 18 * words). The rows * columns * directions represent 3 of the four nested for loops. 18 represents checking the lengths of the directions from 3 to 21. This number is not significant, but Il include it in the attempt to be blunt. The words regards the search of the matching word in the dictionary. My program involves searching every element of the table, rather than using the key due to linear collision.

I can only return the runtime for the 250x250 word grid due to how incredibly long it took to run the file. As of the time of writing this (8:56 pm), the file has been running for 16 minutes and has just completed 2 directions. As of 9.20 pm, it is running the SW direction. As of 9:34, the SW direction is finished with 2 directions left to go. As of 9:58 pm, it is complete. Total reported runtime is 4588.91 seconds.