Boris Topalov, bnt4yb, 3/3/2020

I didn't think implementing the word finder was super difficult. The one thing I remember being caught up on was figuring out when to switch directions. However, it also took me some time to figure out how to implement the hash table. I was struggling to figure out how to make it a proper size; I think the method I did is a pretty good way of doing it- just calling the constructor after knowing how big the hash table should be. I was initially initializing the hash table before that and then resizing it with the proper size, but I think simply initializing it with the proper size all in one step is a better way of doing it.

I've already worked on optimizing my code at this point and have gotten the 300x300 grid output time down to about half a second. I realize I probably shouldn't have done this but I didn't want to deal with writing the shell script so I just continued on working on my word search code. When I first started got the prelab working, the time for my 300x300 grid was around 50 seconds. After compiling with the -O2 flag, runtimes became significantly faster- almost 50% faster, in fact.

Additionally, the runtimes of the smaller grids didn't change that much when optimizing. I was using the 4x7 grid to check the runtimes if they were getting faster since it took much less time than using the bigger grids. However, the time differences in the runtimes between my first change and second change were pretty small. This is probably just because it runs very quickly with small inputs regardless of the level of optimization, which relates to the big theta run-time, as we only look at big inputs when analyzing runtimes.

The big theta runtime of my word search is theta(r*c*w). I have two for loops that depend on input size (one loop for the number of rows and another nested for loop for the number of columns). Within those loops, I am calling find on each word in the grid. The worst case runtime for this would be w, since it's a lookup in a hash table and all of our elements could hash to the same spot so our hash table would basically be a linked list.

I found the shell scripting tedious at times because the syntax is so different than other programming languages I am used to. Also, the spacing is very specific and you have to do things like put two parentheses around arithmetic expressions, which just made it annoying when trying to figure out why my script wasn't working properly. I'm sure that scripts can be very useful and have read that they can automate a lot of tasks, but I haven't been exposed to enough of them to understand and grasp the extent of their usefulness.