1. I did have to change the output of my code so that the entire grid was not printed to the console. I also changed some of the syntax when printing the words and direction so that it matched exactly the example text files. Other than that, my program had no differences from the example text files.
2. Without the -O2 flag, my program ran the 250x250 in 11.319395 seconds when printing to the console. With the -O2 flag, my program ran the 250x250 in 8.126260 seconds when printing to the console. It was about 3 seconds faster.
3. My program ran the 250x250 in 8.126260 seconds when printing to the console. My program ran the 300x300 in 5.719543. I ran this on my personal laptop.
4. Big Theta here is r*c*w*8 because the search algorithm runs in constant time and is dependent on the number of letters in the grid (defined by the rows and columns), the constant, max length of a word, and the 8 possible directions for the search to go in the grid.
5. It took me a little while to figure out how to declare an array as a private member of a class and then initialize it with a length that is not known beforehand. Creating the constructors were probably the hardest part of building the hashTable class. I spent a lot of time using the debugger to fix my hashTable class after I thought I had created a good implementation. Reading in the two text files took a little getting used to because I was unfamiliar with files as input in C++. The quad-nested for loop took a while to wrap my head around and the first time I tried my code it was very slow because I did not put any restrictions on how long the words could be. I spent a good bit of time tweaking the code to get the word search optimized for the problem.
6. The shell scripting part I found kind of odd. The spacing caused me a lot of problems early on. I also don’t like how it does arithmetic expressions, I find it cumbersome. Setting command line input as variable I also thought was a little counter-intuitive. In general, I just don’t really like how variables are created and then used. They seem very useful but kind of cumbersome.