Josh Martin

Professor Snehasis Mukhopadhyay

CSCI 36200 Data Structures

15 February 2018

Knight’s Tour

In my Data Structures class, we did a programming assignment called the “Knight’s Tour”, this problem is as follows:

You have only a knight’s chess piece and you must traverse every position on a chess board and only visiting each position on the board, once.

For the first 32 moves, we got to use an algorithm called “ Warnsdorff’s Rule”. Which calculates the move with the fewest next moves then moves to the one with the fewest and repeats until it finds a solution. Warnsdorff’s Rule works pretty well, it doesn’t have to backtrack that often. However, it took me a while to figure out what was going on. There is not much good documentation on this Algorithm. After doing Warnsdorff’s Rule, doing a exhaustive search on the rest open positions on the chess board was pretty easy.

Overall, this assignment was not hard. It was much easier then the homework (way too hard because of the proofs of big O notation). I was able to finish this project within 3 days. However, I spent most of my time just fixing memory leaks. I still haven’t been able to fix one block of 32 bytes, because of a pointer on my postion class, I am unsure on how to fix it.

Like I said before, this assignment was pretty easy. I tried my best to organize my code well by putting files into folders, based off of their use case and inheritance. I also tried to create clear function and variable names. Lastly, I also tried for the best complexity, I could personally achieve. I also found a function on stackoverflow, that allowed me to format strings nicely. I had no problems implementing link lists and stacks into my project. I felt that understanding link lists and stacks was the easiest part of my project. This might be because I have taking classes before this one that talked a lot about stacks and link lists.