Game Readme

This was the most tedious part of the assignment. To start, I had to write out the method in C to know what is going on. Next I converted it into assembly code which was the hardest part of the assignment. First, I had to malloc space for a new board to make compared to the old board. Then I had to run 2 for loops to go though each cell and check its neighbors. This was very tedious to do in assembly and took a very long time because before you can perform a check on 1 of the eight possible neighbors you have to make sure you are staying within the bounds of the array or you will get a segmentation fault. Also, using the loop instruction was sort of confusing because of how the program is designed and using the loop while already having a for loop going through the rows was very difficult. The big O analysis of this was O(N^2). Now after creating the new board, I set the old boards pointer equal to the new boards pointer and free'd the old board in memory. After all of this, the Big O runtime of my update method was O(N^2).

Next I had to build the makefile. This was also kind of hard because I did not know at first how to link the .s and .c files together. However, I did eventually figure it out and now know how to link x86 code to my C code.