

Christopher Gundlach Homework 8

In building and choosing my algorithm one of the important things I learned about Light up is that each puzzle has a unique solution: ie there is only one way to satisfy the numbers on the black cells and at the same time to illuminate all the white cells with no bulb shining on another. I tested this repeatedly on several puzzles at <https://www.puzzle-light-up.com/>. This led me to realize that once a player had placed all their lights it could be compared to the correct position, similar to how the online game went. The solution was never checked until the player has placed all their lights. I chose to use brute force to compare the answers with the selections of the player. I created vectors of each using a struct that held the coordinates, so I used a brute force method to compare every user entry to the first entry in the solution, then the second, and so on, breaking out when a match was found. This required two for loops an if statement and iteration for a counter. This led to a runtime of my algorithm of $O(n^2)$. It could easily be scaled up for any sized board.

I can also discuss how the game itself is NP complete. Based on the decision problem of whether there is a solution for a given board, this can easily be found in polynomial time, as seen above. That puts the game in NP. It is also N hard because you can reduce CircuitSAT to Light Up, per McPhail (2005). He specifically builds multiple boards that functions the same way as circuits in CircuitSAT. It is rather interesting and impressive.