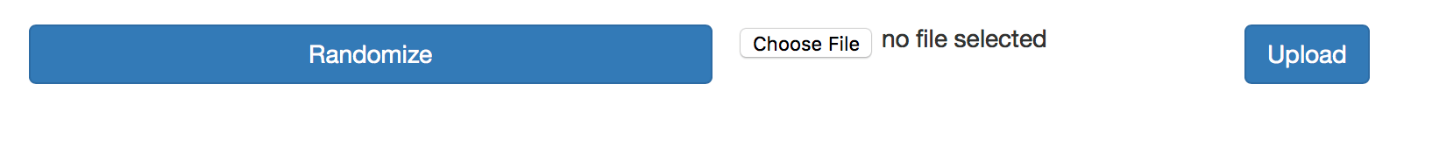
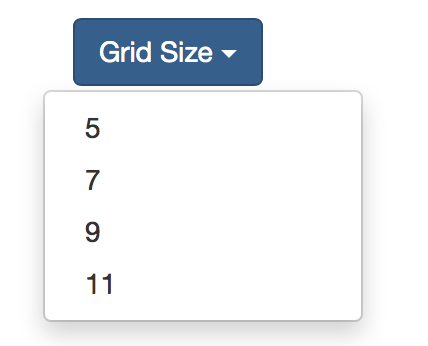
CS 440: Assignment 1

Task 1: Puzzle Representation

The structure of the application and GUI are built on Python 3, and Flask which is a Microframework for web development. We decided to use python because it’s ability to quickly build applications with its simple syntax, and powerful libraries. Building a website for the GUI allows us to ensure display compatibility across systems. Each time the page is loaded a random puzzle is generated based on the chosen dimension in the GUI. The valid sizes for the puzzle are n=5,7,9, or 11, however other sizes could be added if necessary. Each cell in the n x n puzzle has a randomized number 1 to n-1 except the goal cell which is in position [n-1,n-1].

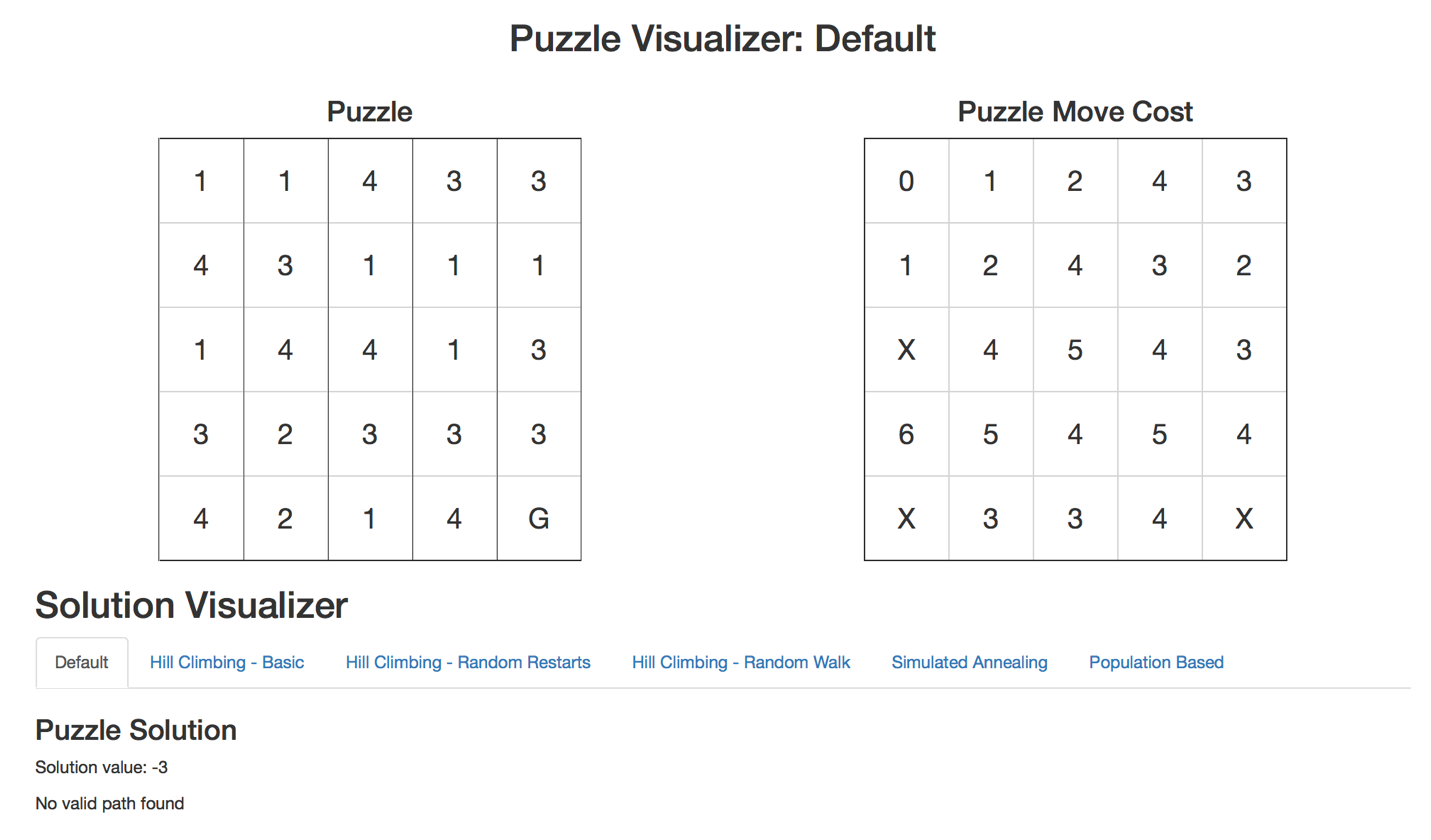


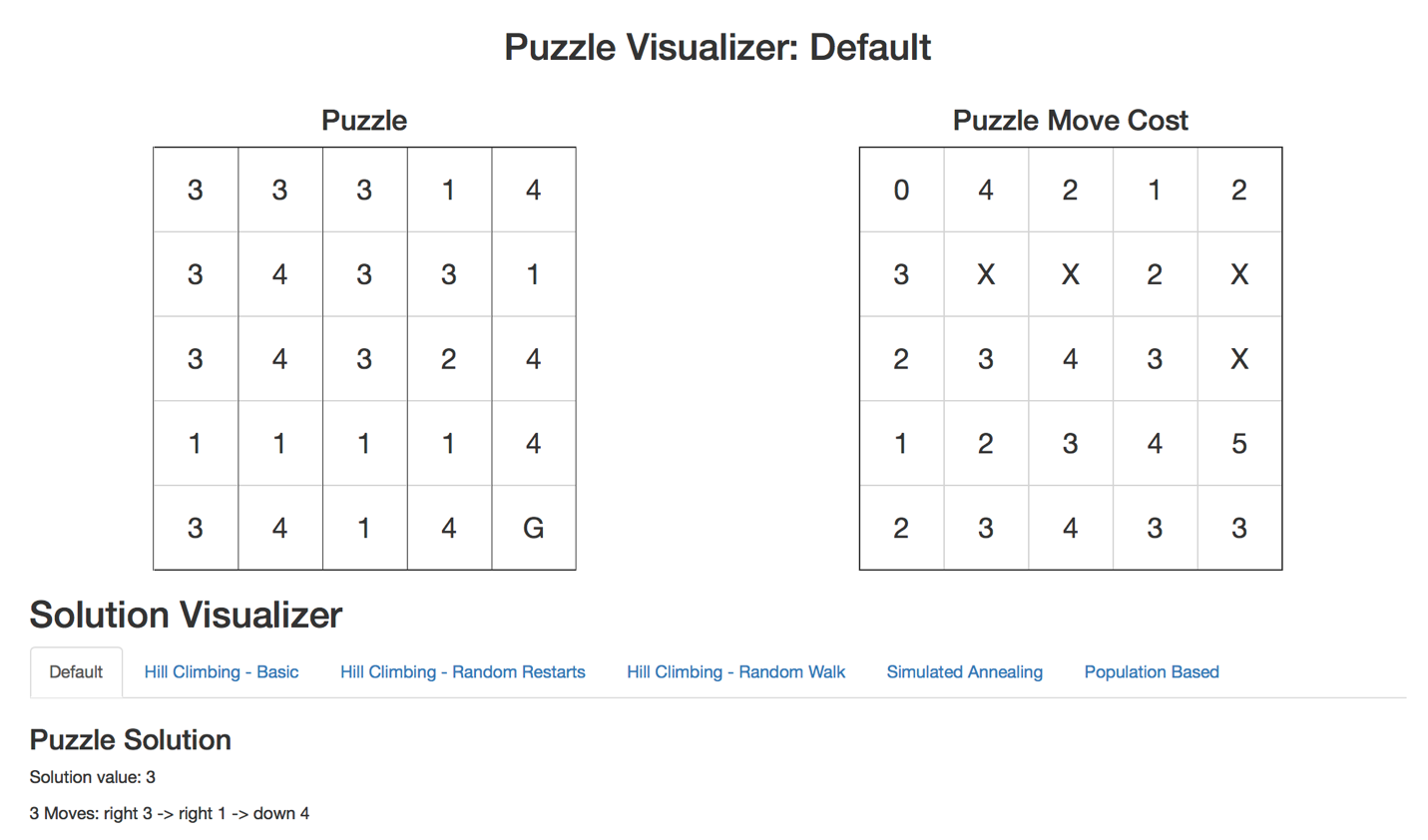
*Include GUI Examples here*

Task 2: Puzzle Evaluation

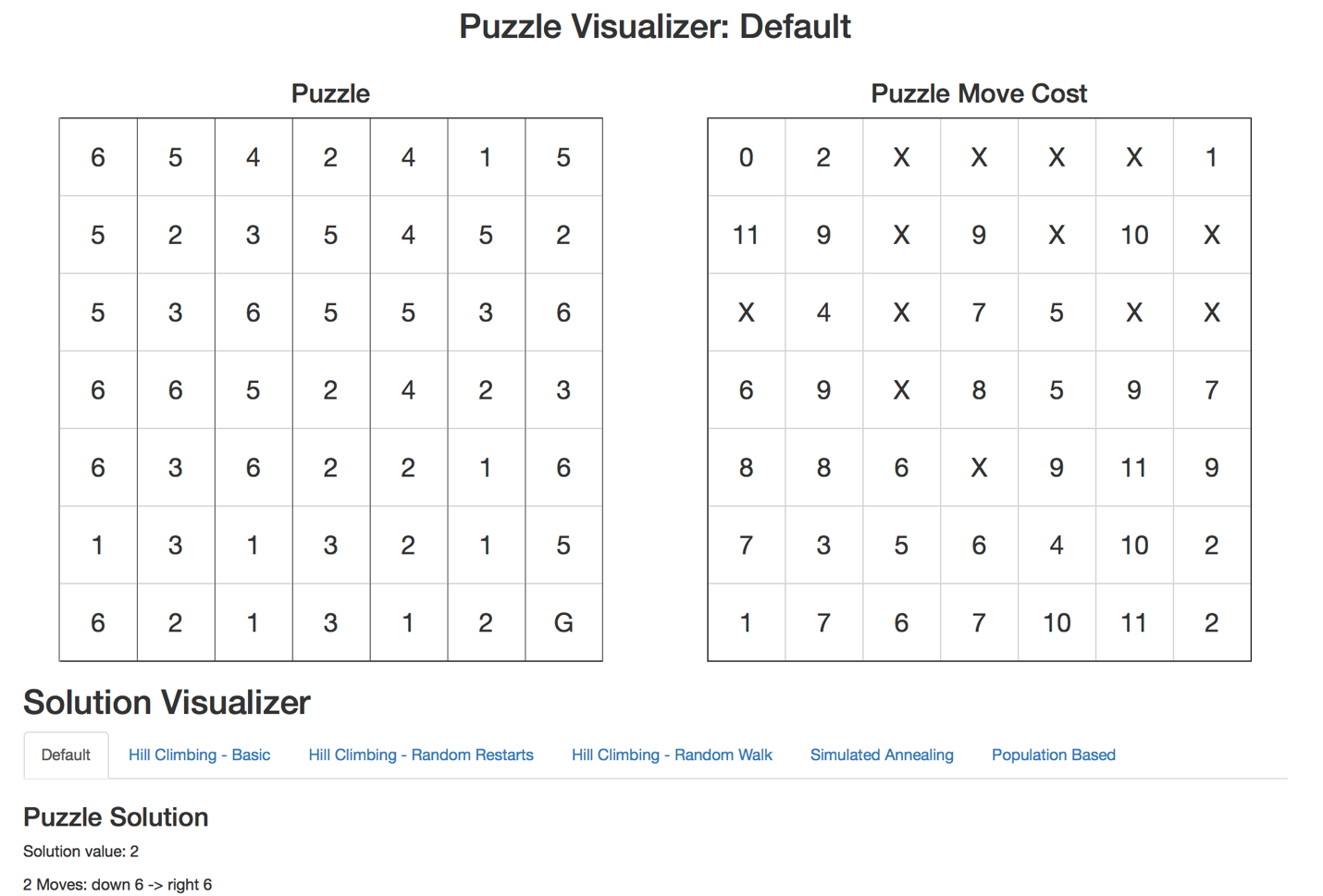
For this task we use a tree type data structure which holds all the unexplored nodes. We then perform a BFS type analysis on the tree to determine the distances from the root each node is located. To keep track of already visited points an auxiliary array of (1,0) size n x n is kept. As children are added to the queue the distance array is populated until we exhaust the entire tree.

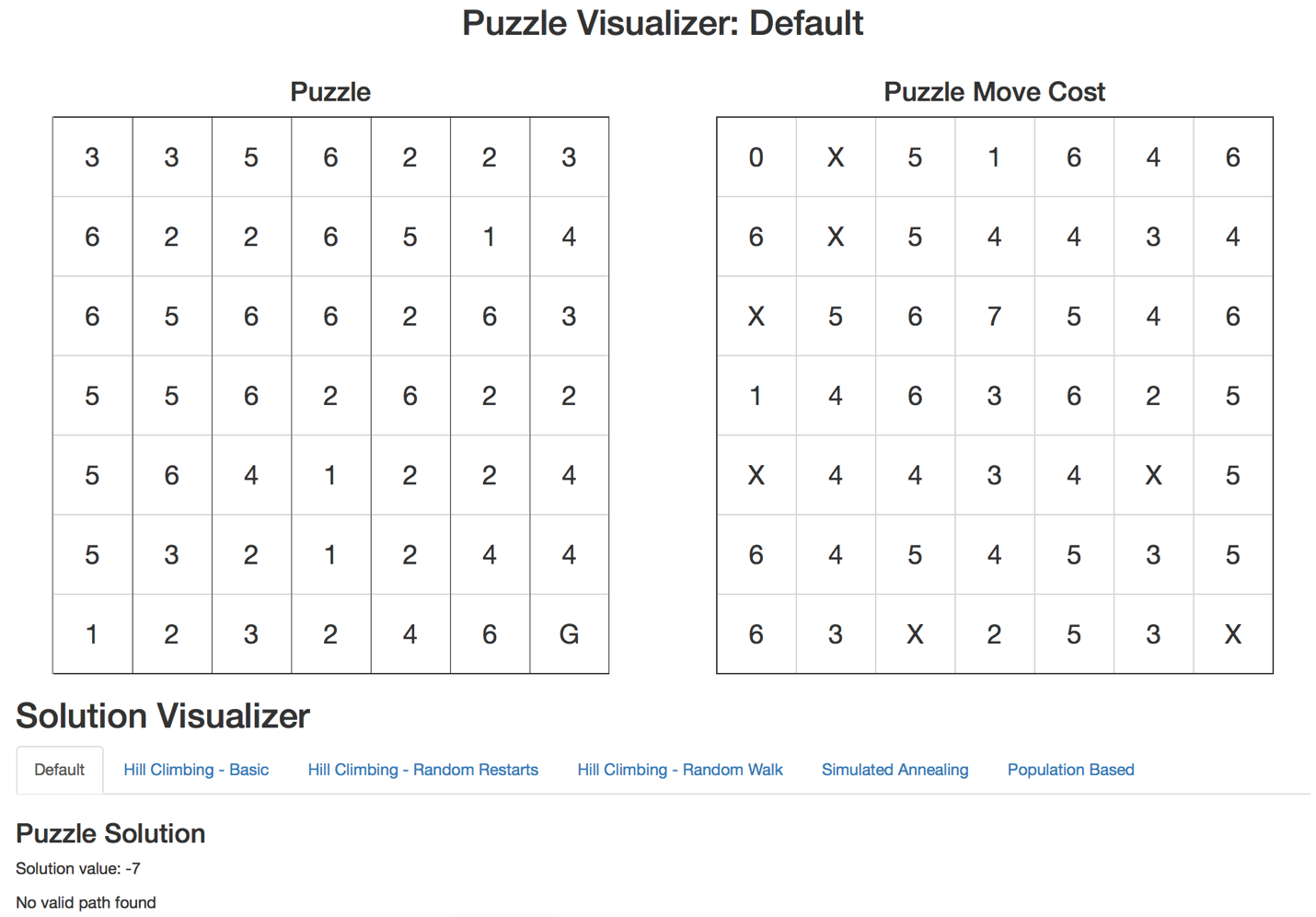
5x5 Puzzles



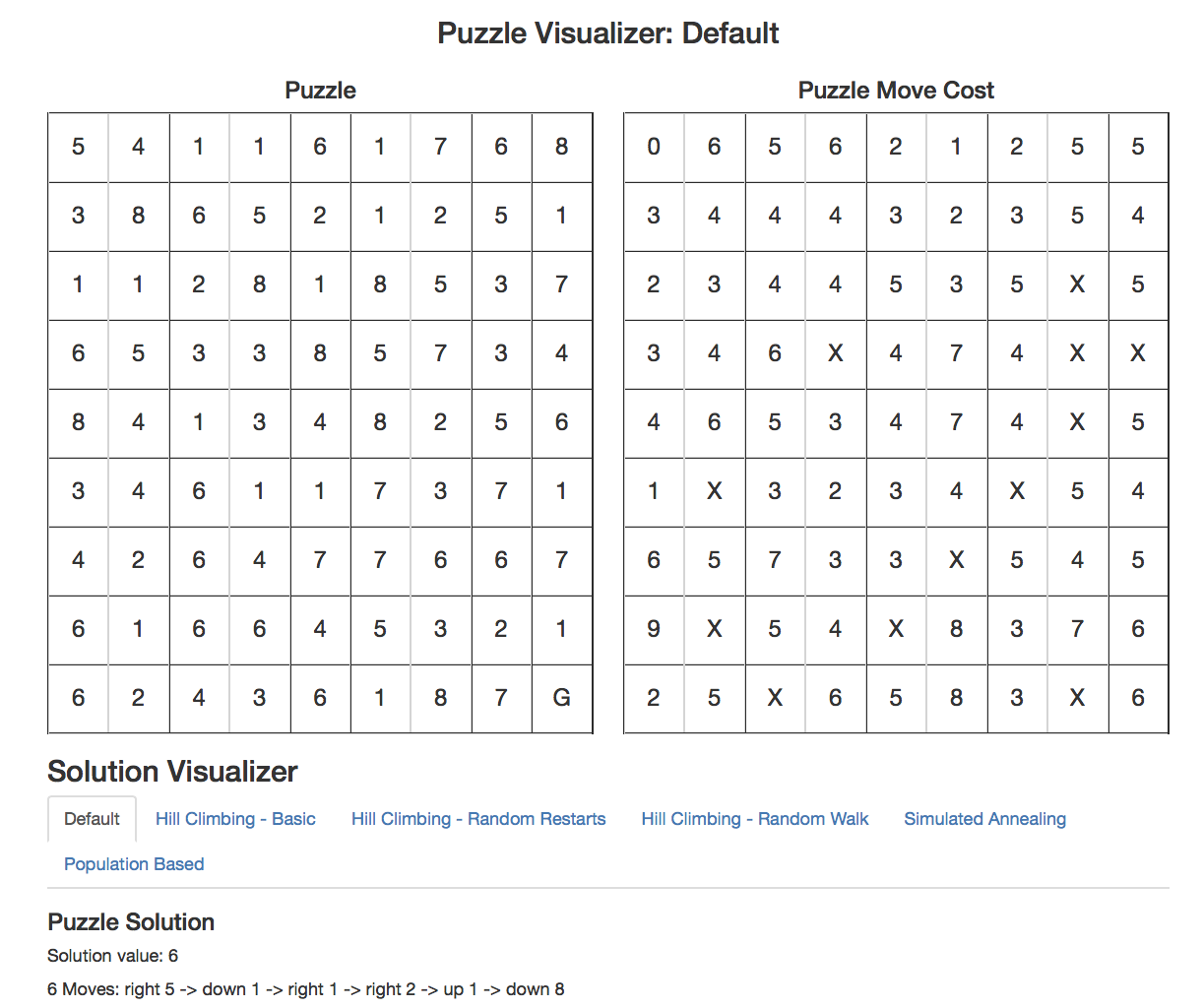


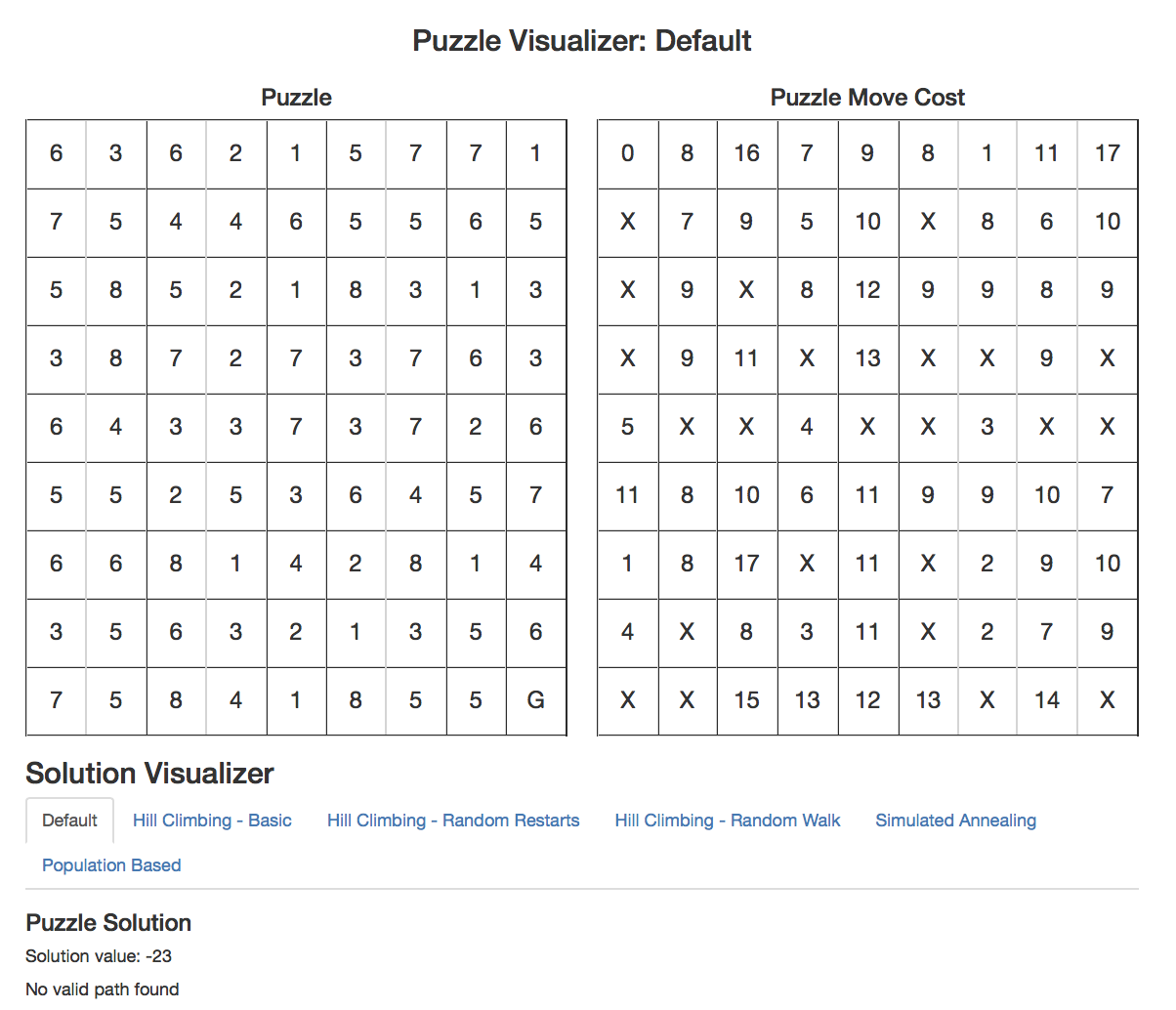
*7x7 Puzzles*



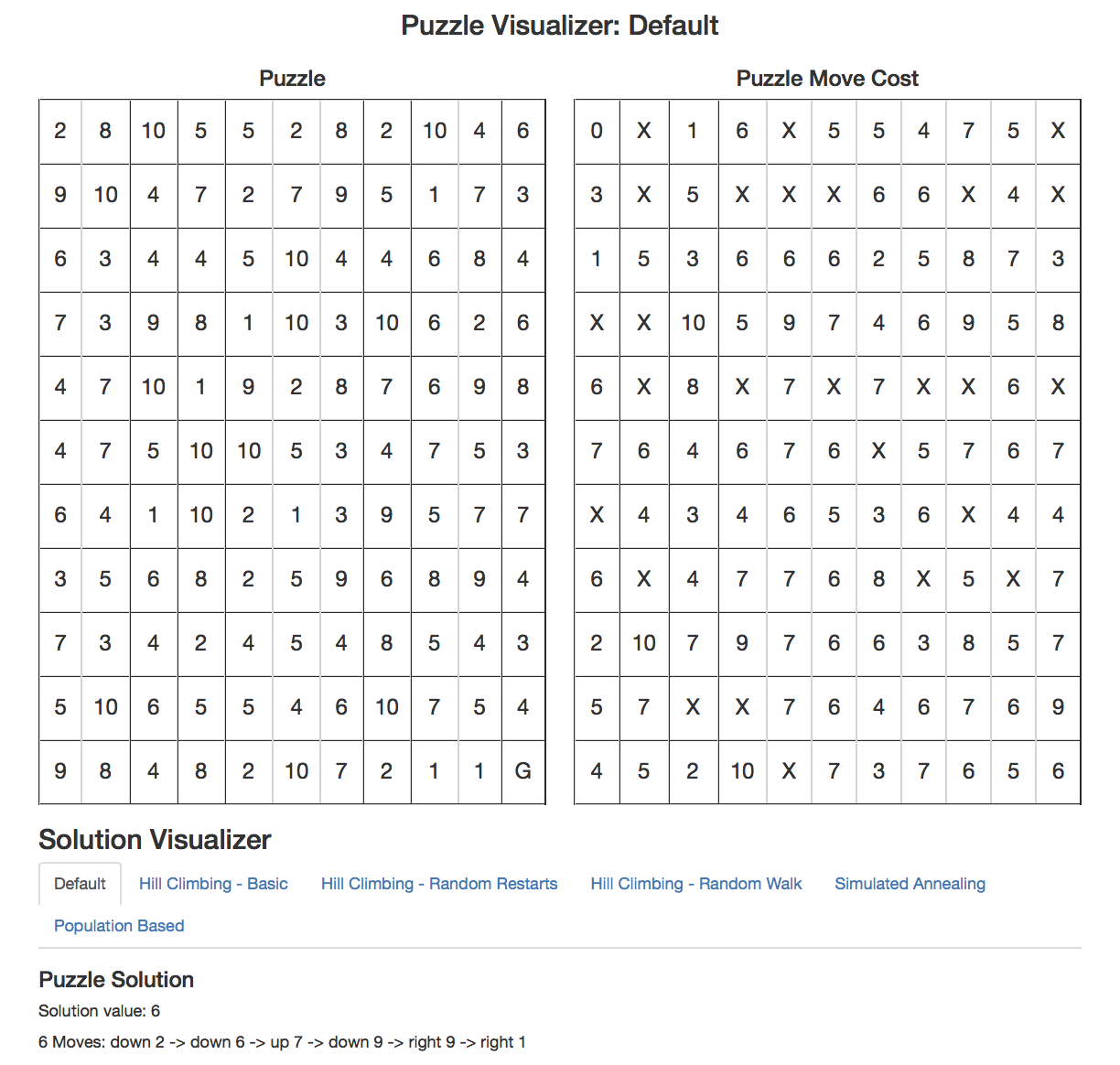


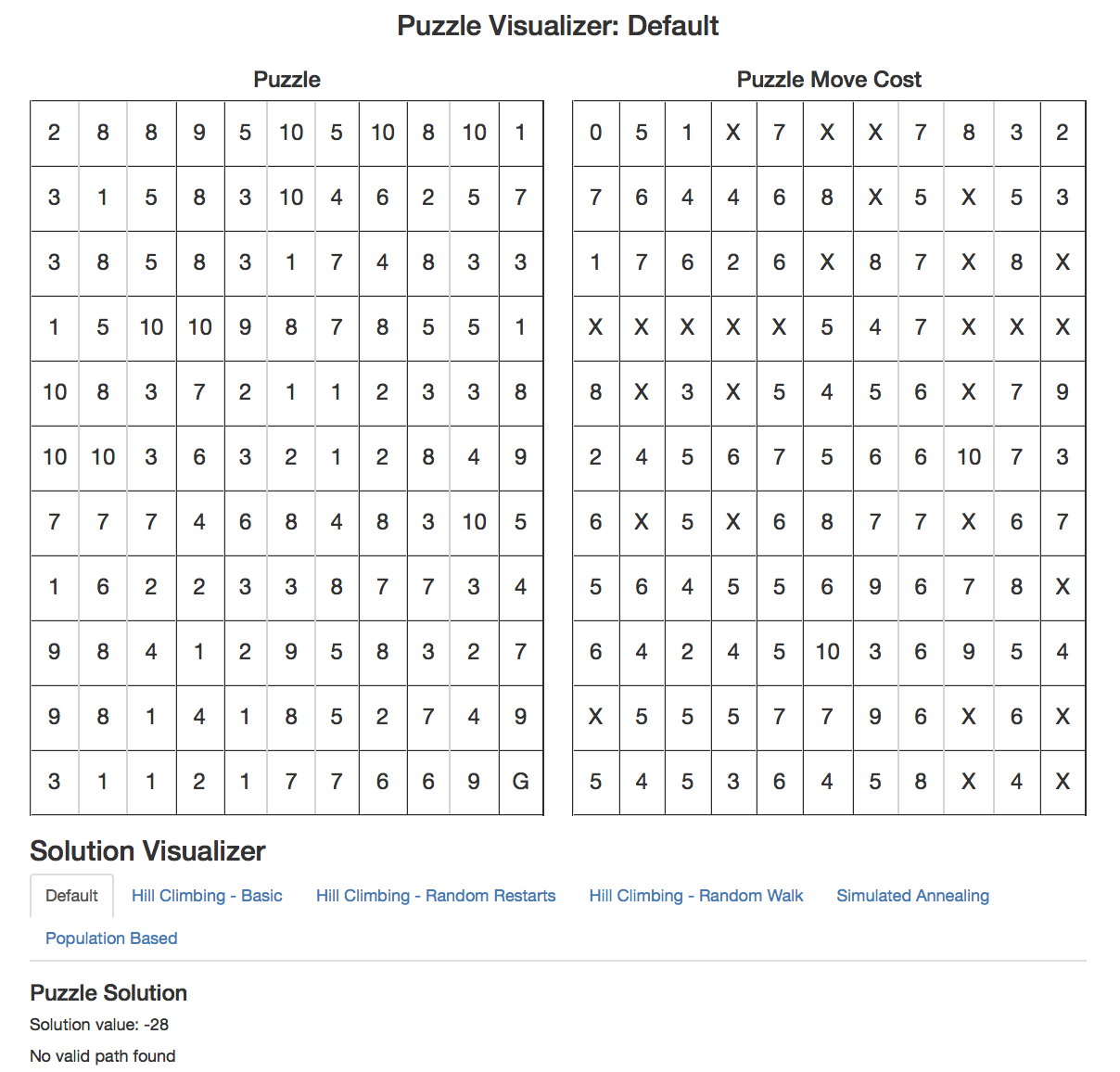
*9x9 Puzzle*





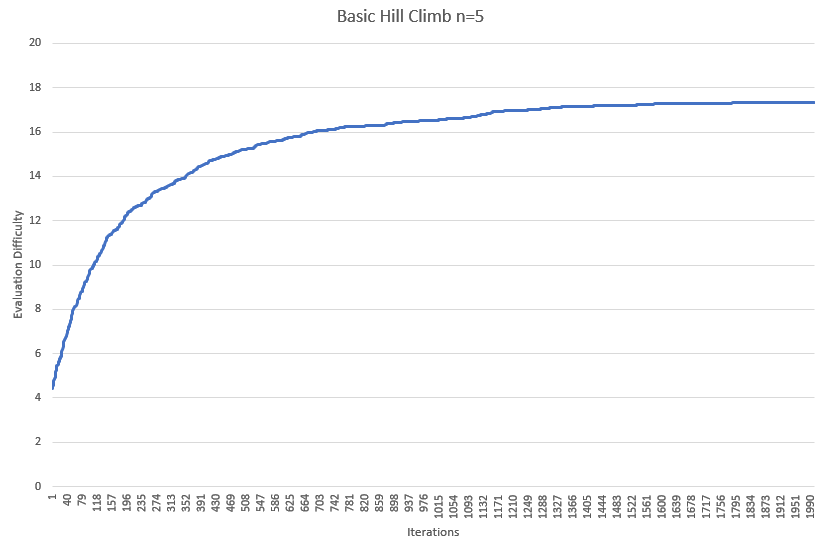
*11x11 Puzzle Size*

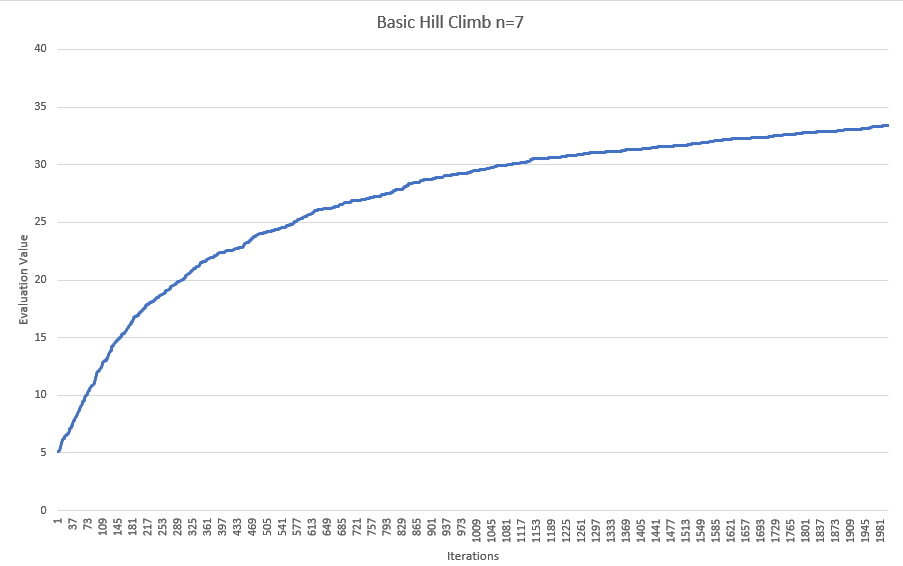


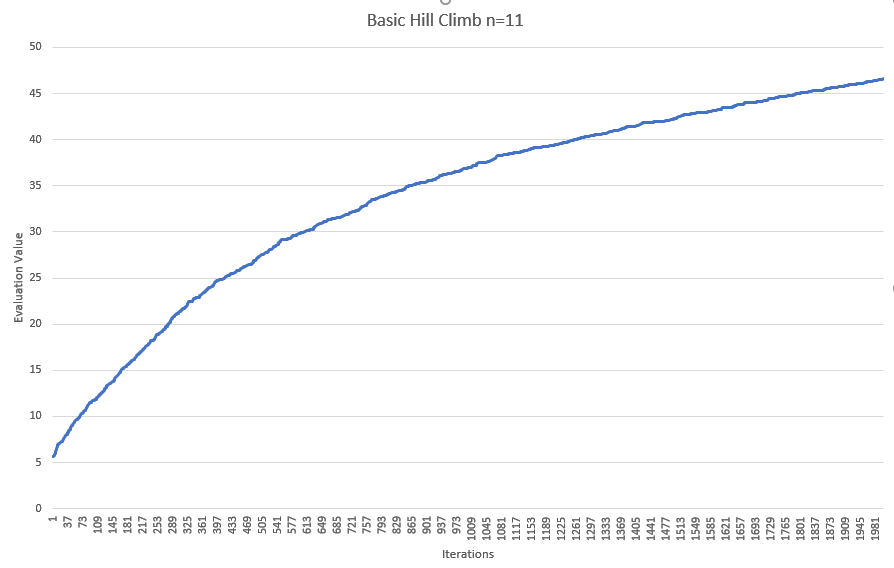


*Put puzzle solutions here*

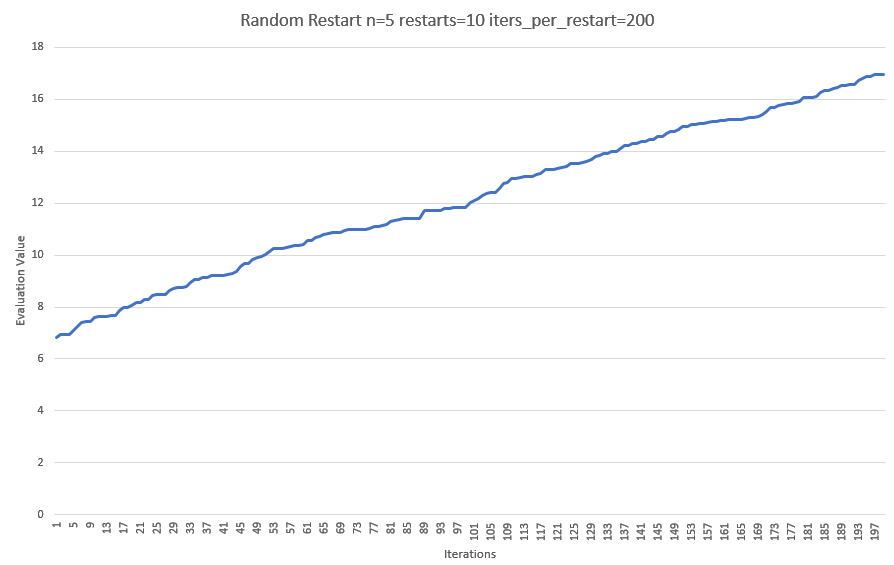
Task 3: Hill Climb Basic

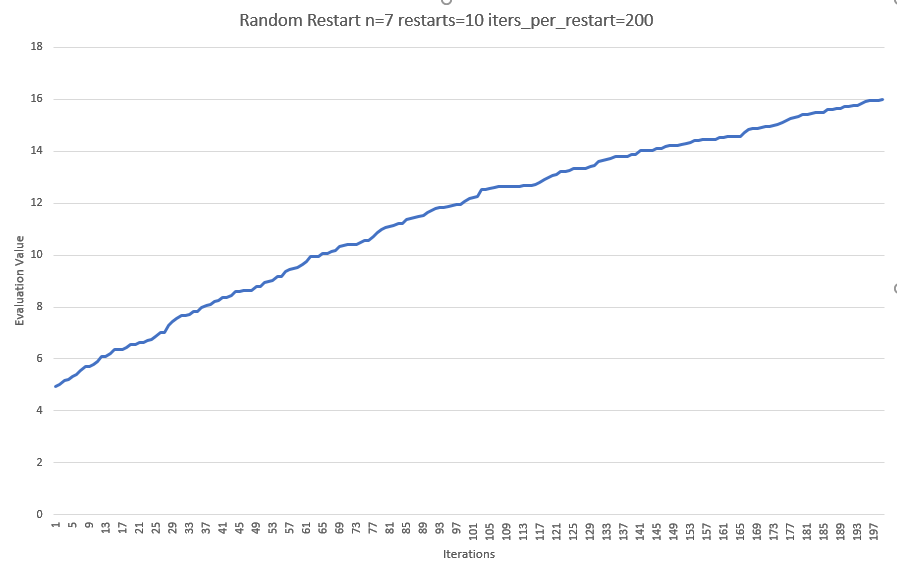


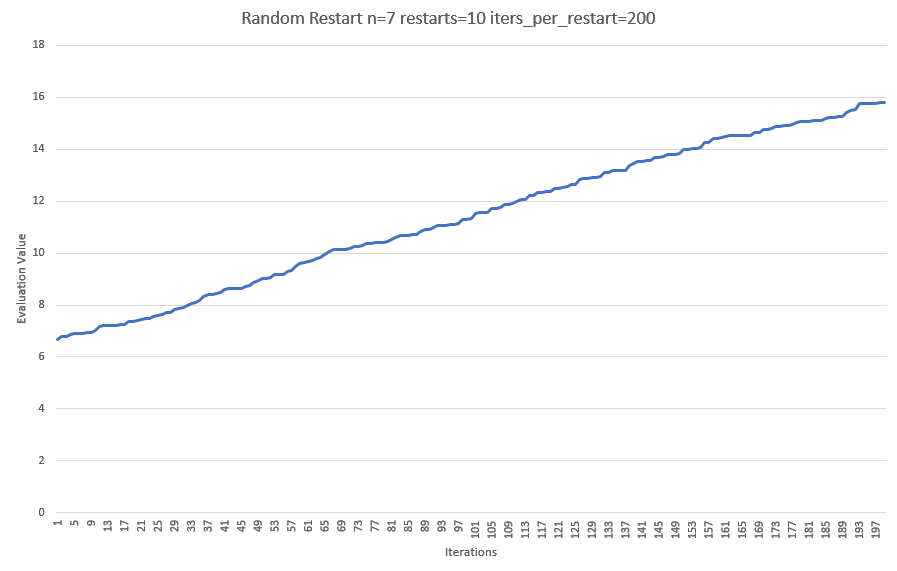


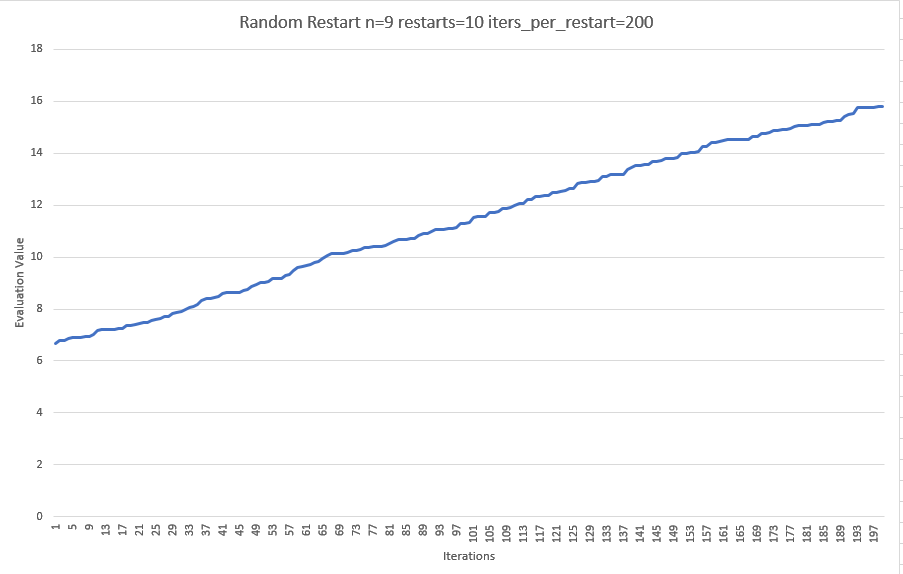


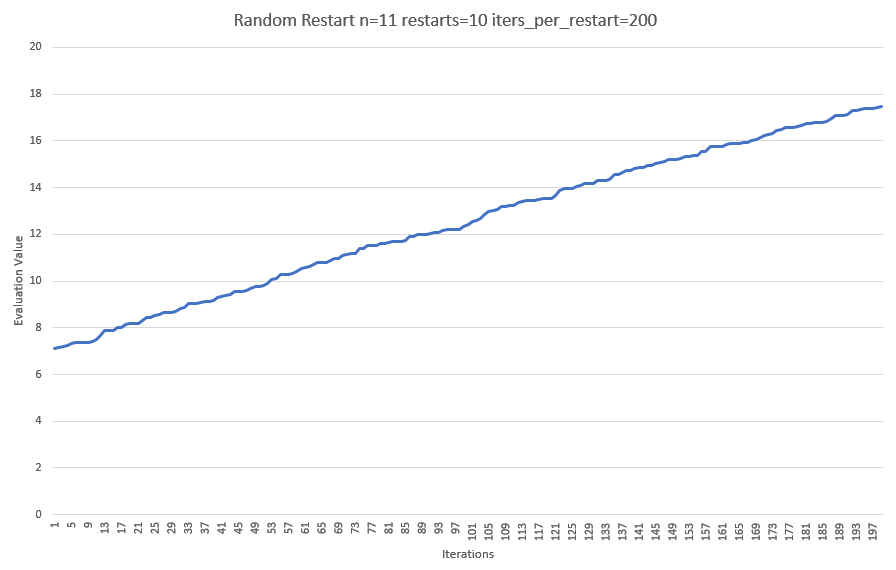
Task 4: Random Restart



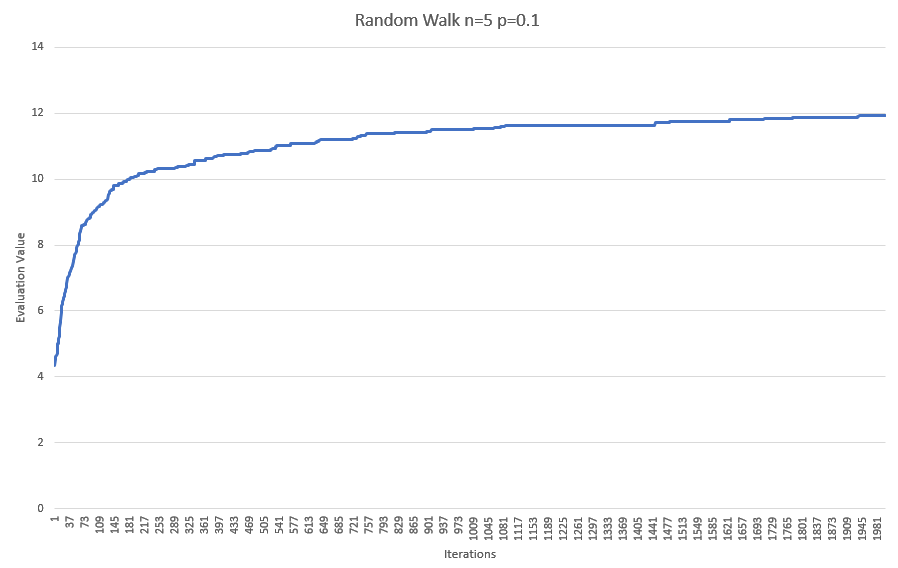


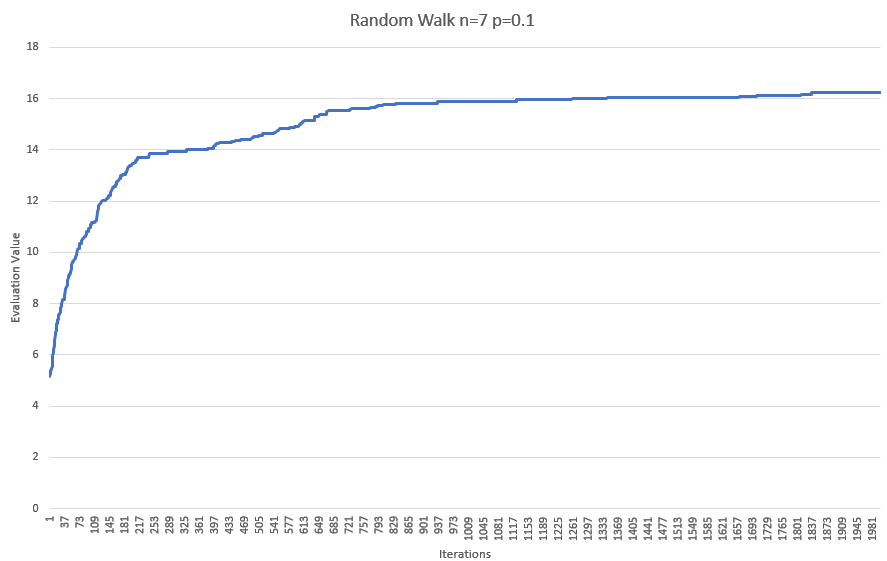


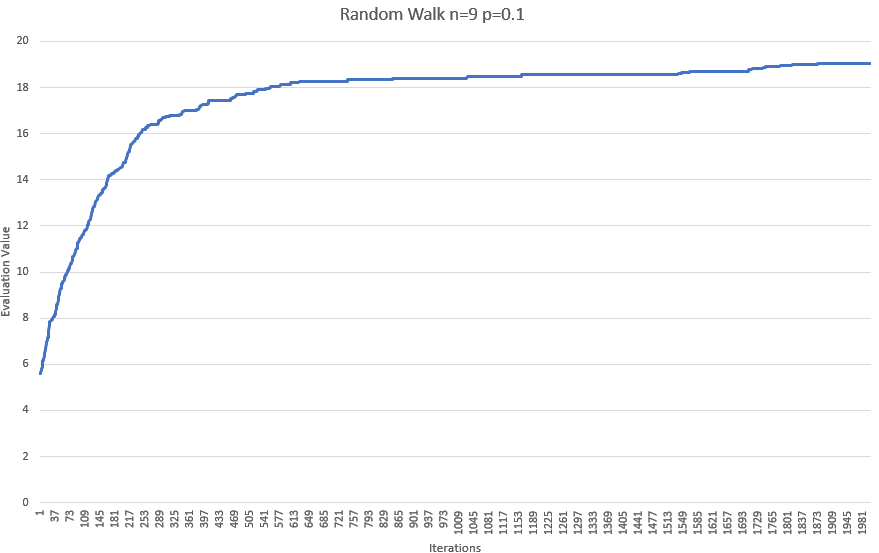


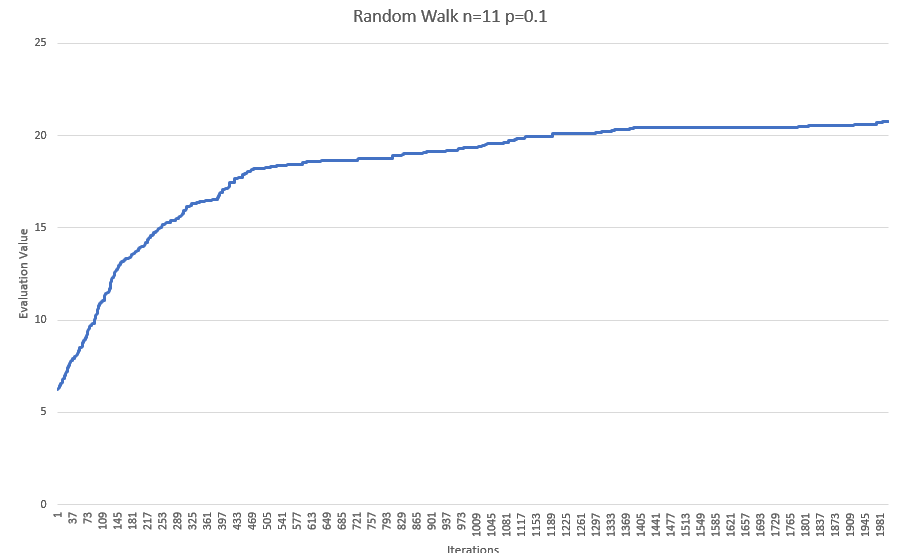


Task 5: Random Walk









Task 6: Simulated Annealing

