A study of Knight's Travails

CS F211 : DATA STRUCTURES AND ALGORITHMS

ABHINAV VERMA	2020A7PS1093H
MUFADDAL JIRUWALA	2020A7PS1720H
SAI HEMANTH ANANTHOJU	2020A7PS0116H

Problem Statement:

We all know of knights in a chess board. We also know of the unique movement of the knight (an “L-shape” to be brief).
We aim to show the shortest possible path that a knight can take to take down a target chess piece, in a N x N chessboard.

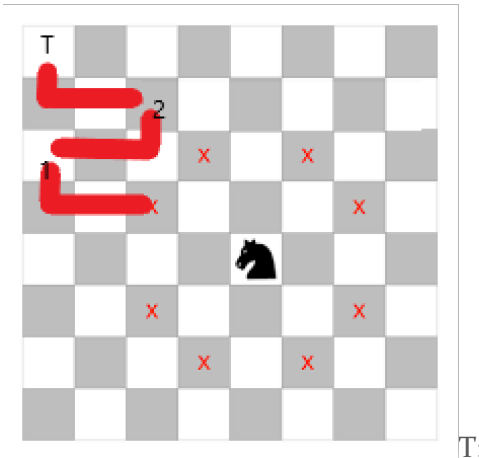
We also give the player the choice to block certain squares with a friendly-pawn, that the knight may choose another path.

DATA STRUCTURES TO BE USED:

The initial approach to this problem would be the finding the shortest path in an unweighted graph.
We take each square in a chess board as a graph node, and the board itself to be the graph.

We aim to use a special form of Breadth First Search (BFS), as we are using an unweighted graph here, and search all the possible paths that the knight can take. Then the algorithm chooses the shortest path.

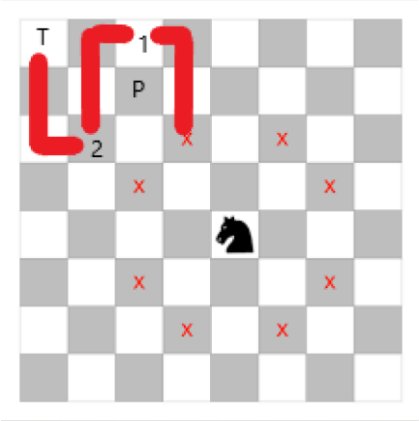
If the player chooses to block the path of the knight with a friendly piece, the knight then removes the particular node from the graph and resumes BFS.



T: Target

In the above diagram, we can see that the minimum steps required by the knight to reach the target is 4 (including the first red cross).

Now, let's say that the path is blocked with a friendly pawn, which is placed at square 2. Hence, the knight changes its path as follows:



T: Target, P: Friendly- Pawn

In this case, the length of the shortest path remains unchanged, and the path changes. However, this might not always be the case, and the length of the shortest path may increase.