CSE 4082 – Assignment 1

Peg solitaire ("Solo Test" in Turkish) is a board game for one player involving the movement of pegs on a board with 33 holes. The shape of the board can be seen in the figure below. Initially, all the holes except the center hole are filled with pegs. A valid move is to jump a peg orthogonally over an adjacent peg into a hole two positions away and then remove the jumped peg. The objective is to empty the entire board except for a solitary peg in the central hole.
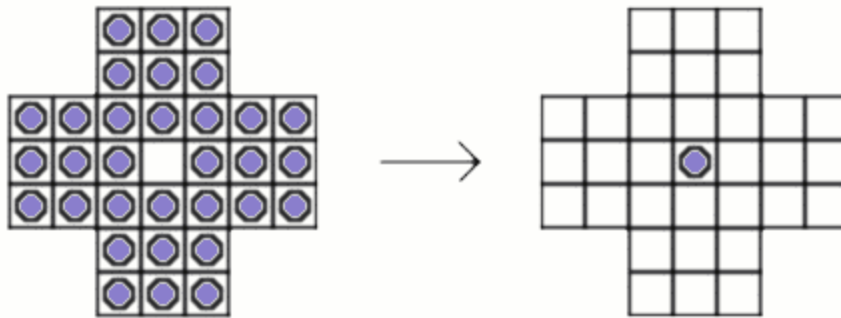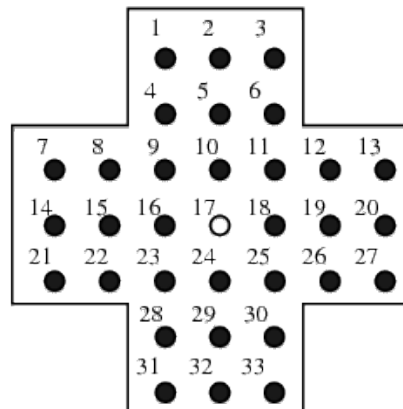
Figure 1 (a) The Initial State   *(b) The Goal State*

The labels of the slots are as follows:

1. Implement a program for solving peg solitaire:
   a. Breadth-First Search
   b. Depth-First Search
   c. Iterative Deepening Search
   d. Depth-First Search with Random Selection
   e. Depth-First Search with a Node Selection Heuristic

Your program should input a search method (a-e) and a time limit (t) value, and print the following information:

i.   The search method and the time limit,
ii. A message indicating the return status: "Optimum solution found.", "Sub-optimum Solution Found with XX remaining pegs", "No solution found – Time Limit", "No solution found – Out of Memory".
iii. The board states from the initial state to the final state using a suitable representation,
iv. The time spent,
v.   The number of nodes expanded during the search.
vi. Max number of nodes stored in the memory during the search.

2. For each search method a-e, report the output of your program using a time limit of 60 minutes.

Implementation Notes:

a. The project should be done in groups of two.
b. For methods a to c, if there are multiple children to be put inside the frontier list, put the children in such an order that the child node with the smallest numbered peg is removed from the board is selected first.
c. For Depth First Search with Node Selection Heuristic, you should design your own node selection heuristic.
d. Your source code should be cleverly commented.
e. Further details of the project will be discussed in the class.
f. Your report should include program outputs and implementation details for each search method a-e.