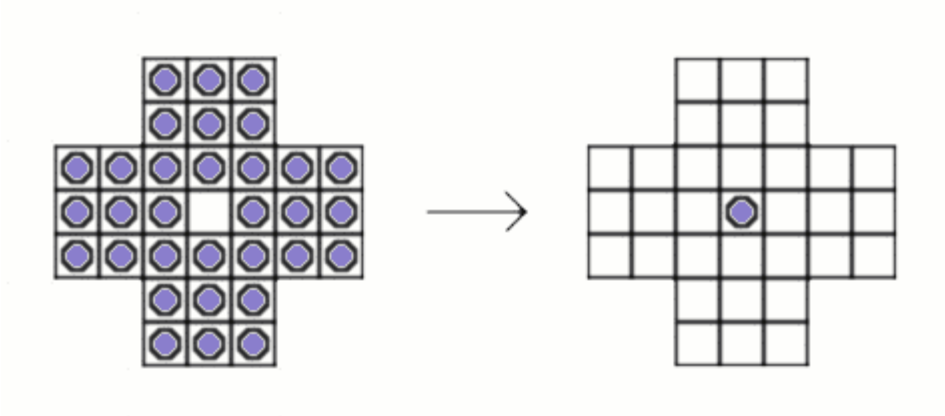
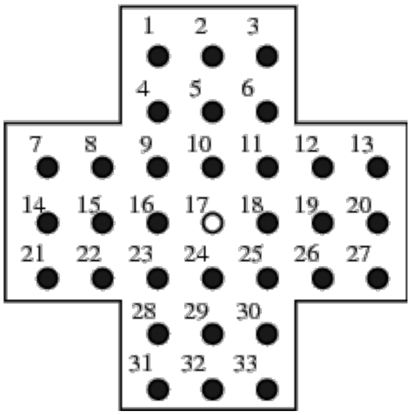
# CSE 4082 – Assignment 1

( Due 05.12.2018 at 23:59, electronic submission only, to cse.cse482@gmail.com )

Peg solitaire ("Solo Test" in Turkish) is a board game for one player involving 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 to remove the jumped peg. The objective is, making valid moves, to empty the entire board except for a solitary peg in the central hole.



The labels of the slots are as follows:



1. Implement a Java/C/C++ 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”.
iii. The optimum solution itself or the sub-optimum solution starting from the initial state and ending in the final state (you should print board after all moves using a suitable representation).
iv. The time spent for finding the optimum solution if the optimum solution is found.
v. The number of nodes expanded (display even if the optimum solution is not found, i.e. timeout occured or memory is consumed).

2. For each search method a-d report the output of your program with a time limit of 60 minutes.

Implementation Notes:

a. 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.
b. For Depth First Search with Node Selection Heuristic, you should design your own node selection heuristic.
c. Your source code should be cleverly commented.
d. Further details of the project will be discussed in the class.