DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
nec, Bhaktapur

ARTIFICIAL INTELLIGENCE
Lab Sheet III

1. Implement automated vacuum cleaner reflex agent.
2. Improve the vacuum cleaning agent's efficiency by implementing a model-based agent. Provide your arguments.
3. WAP in Python will implement DFS/BFS on the water jug problem.
   Given a 4 - litre jug filled with water & an empty 3 - litre Jug, how can one obtain exactly 2 liters in 4 litres jug. There is no measuring mark on any of them.
   - Define WaterJug Class with a constructor to initialize the initial and goal state
   - Define boolean goalTest(current_state, goal_state) to check if current_state is goal_state or not
   - Define successor() with reference to the production rules to generate possible child(s).
   - Verify if successor() is working properly or not
   - Define DFS/BFS search algorithm to find the solution
   - Modify search algorithm to store state,parent in CLOSED list and also define generate_path() to provide the path of solution.

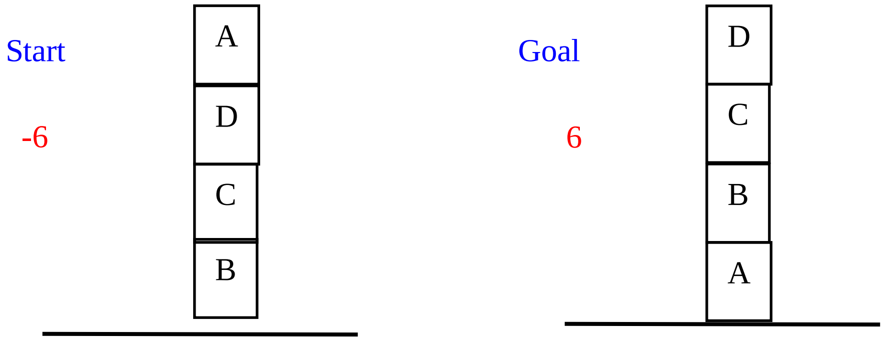
4. Based on your last digit of your CRN, implement the following search problems as above

The last Digit of CRN	**Problem**
0	City-Map Problem
1	n-Puzzle
2	Missionaries and Cannibals
3	Towers of Hanoi
4	Tic-Tac-Toe
5	Block World
6	Man Goat Lion Cabbage
7	Monkey and Bananas Problem
8	n-Queen Problem
9	Water Jug with arbitrary size, i.e., the user will input the container size. Also update the goalTest() to verify the description instead of a fixed goal_state(2,0).

5. WAP in Python to calculate the heuristic value of the states for Blocks World Problem as follows

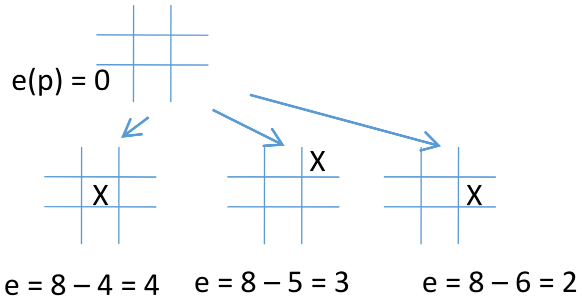
Global heuristic: e(p) is calculated as

- For each block that has the correct support structure, give +1 to every block in the support structure.
- For each block that has a wrong support structure: -1 to every block in the support structure



Blocks World

6. WAP in Python to calculate the heuristic value of the states for Tic-Tac-Toe as follows



Heuristic function:
e(p)= No. of complete rows, columns or diagonals are still open for player) – (No. of complete rows, columns or diagonals are still open for opponent)

7. Solve the 8 puzzle problems using A* algorithm in Python.
8. Write a program to implement the steepest ascent hill climbing for the 8-puzzle problem. Develop appropriate heuristic functions.
9. WAP to demonstrate the effect of temperature on the probability of choosing an inferior node by selecting an appropriate temperature schedule.