Exam No: _____

GANPAT UNIVERSITY
B. TECH SEM-VI (CE/IT/CE-AI)
FIRST INTERNAL EXAMINATION – MARCH 2024
2CEIT602: Artificial Intelligence

TIME: 1 Hour **TOTAL MARKS: 20**

Instructions:
1) Figures to the right indicate full marks.
2) Be precise and to the point in your answer.
3) Open Book Exam-Following Materials in hard copies only, will be allowed to carry.
   - Textbooks/Reference books
   - Photocopy of content from the reference books
   - Lecture notes prepared by student
4) The text just below marks indicates the Course Outcomes Numbers, (CO) followed by the bloom's taxonomy level of the question, i.e., R: Remembering, U: Understanding, A: Applying, N: Analyzing, E: Evaluating, C: Creating.

Q.1 Given the truth tables of three Boolean functions with two inputs (x1 and x2), create the McCulloch-Pitts (MP) neuron models to represent and learn these functions. If the MP neuron model cannot be created for any of the Boolean functions, for those functions, explain why the MP neuron model fails to represent them, providing clear reasoning. **[05] 3A**

Boolean Function-1

x1	x2	Output
0	0	1
0	1	0
1	0	1
1	1	0

Boolean Function-2

x1	x2	Output
0	0	1
0	1	1
1	0	0
1	1	0

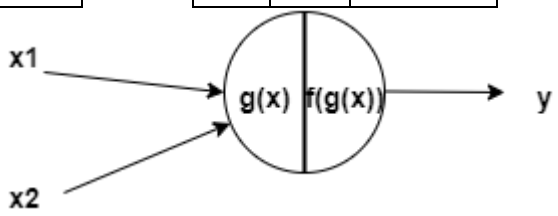
Boolean Function-3

x1	x2	Output
0	0	0
0	1	0
1	0	0
1	1	1

Solution:

MP Neuron:

$$g(x) = \sum_{i=1}^{n} x_i \qquad y = 1, g(x) \geq \theta \qquad y = 0, g(x) < \theta$$



If (x1 + x2) $\geq \theta$, the output is 1; otherwise, the output is 0.

Boolean Function-1:

x1	x2	Output	g(x)	Conditions
0	0	1	0	$0 \geq \theta$
0	1	0	1	$1 < \theta$
1	0	1	1	$1 \geq \theta$
1	1	0	2	$2 < \theta$

For the first row (x1 = 0, x2 = 0, output = 1), we need: $\theta \leq 0$. Therefore, the threshold should be negative or zero. However, for the fourth row (x1 = 1, x2 = 1, output = 0), we need: $\theta > 2$. Therefore, the threshold should be greater than 2. These conditions on the threshold value contradict each other, making it impossible to find a single threshold

value that satisfies all the equations simultaneously. The Boolean Function-1 cannot be represented by the MP neuron model.

Boolean Function-2:

x1	x2	Output	g(x)	Conditions
0	0	1	0	$0 \geq \theta$
0	1	1	1	$1 \geq \theta$
1	0	0	1	$1 < \theta$
1	1	0	2	$2 < \theta$

For the first row (x1 = 0, x2 = 0, output = 1), we need: $\theta \leq 0$. Therefore, the threshold should be negative or zero. However, for the fourth row (x1 = 1, x2 = 1, output = 0), we need: $\theta > 2$. Therefore, the threshold should be greater than 2. These conditions on the threshold value contradict each other, making it impossible to find a single threshold value that satisfies all the equations simultaneously. The Boolean Function-2 cannot be represented by the MP neuron model.

Boolean Function-3:

x1	x2	Output	g(x)	Conditions
0	0	0	0	$0 < \theta$
0	1	0	1	$1 < \theta$
1	0	0	1	$1 < \theta$
1	1	1	2	$2 \geq \theta$

$\theta = 2$ satisfies all conditions calculated in above table. By setting the threshold value to 2, the MP neuron model correctly represents Boolean Function-3, which is the logical AND operation. So, with a threshold value of 2, the MP neuron model can represent Boolean Function-3.

Q.2 Formulate the N Queen Problem as a state space search problem by defining state, initial state, goal state and action. Draw a 4 Queens search tree up to level 1, showing the successor states generated from the initial state. **[05] 1U**

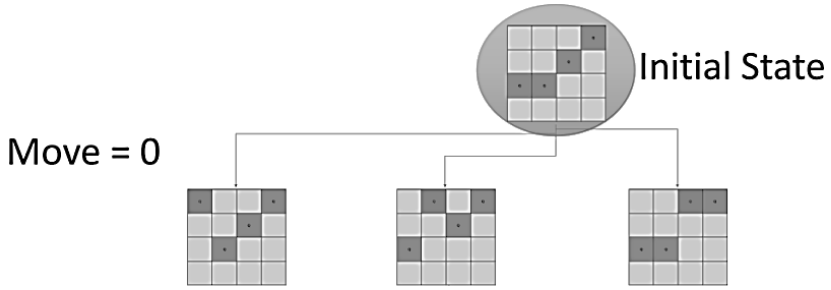
Solution:

States: Exactly one queen is placed in each column and Any arrangement of queens in row on the board

Initial state: Random Arrangements of Queen

Action: Add a queen to the (Move=k)th row.

Goal State: N queens on the board, none are attacked



Q.3 A person starts their journey from city A and needs to visit cities B, C, D, and E exactly once, in any order. All the cities are connected to each other, meaning the person can travel from any city to any other city. Assume that the cities are represented by single letters (A, B, C, D, E), and the initial state and goal state are as defined below. **[05] 4A**

The problem is represented as follows:

- **State:** A pair consisting of the current path of visited cities and the list of remaining unvisited cities.
- **Initial state:** The person starts from city A, so the initial state is ([A], [B, C, D, E]).
- **Goal State:** The goal is reached when the person has visited all cities exactly once so there will be more than one Goal states. So, one of the goal states is ([A, B, C, D, E], []).
- **Action:** At each step, the person can choose any remaining unvisited city randomly and add it to their current path.

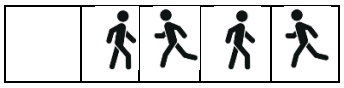
Write pseudocode to solve this problem using Breadth-First Search (BFS) algorithm. Pseudocode should include these components: a class for the state, function generate successors, function check the goal state, and function the main BFS algorithm.

Solution:

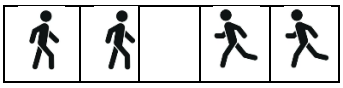
```
class Node:
    Constructor Node(path, remaining):
        Set path = path
        Set remaining = remaining

function generate_successors(state):
    successors = []
    Randomly shuffle state.remaining
    For each city in state.remaining:
        Create new_path by copying state.path and appending city
        Create new_remaining by copying state.remaining and removing city
        Append node(new_path, new_remaining) to successors
    Return successors

Function is_goal_state(state):
    Return True if length of state.remaining is 0, else False

function bfs(initial_state):
    Create an empty queue and enqueue initial_state
    While queue is not empty:
        Dequeue state from queue
        If is_goal_state(state) returns True:
            Return state
        Extend queue with the successors of state generated by
generate_successors
    Return None if no solution found

Set initial_path to ['A']
Set remaining_cities to ['B', 'C', 'D', 'E']
Set initial_state to Node(initial_path, remaining_cities)
Set solution to bfs(initial_state)
If solution is not None:
    Print solution path
Else:
    Print "No solution found."
```

Q.4 The sliding-tile puzzle consists of two walking persons, two running persons, and an empty space in the configurations shown below: **[05] 1A**

Initial Position



Goal Position



State Representation: Consider 'W' letter as a walking person, 'R' letter as a running person, and '_' represents an empty space.

Initial State: _WRWR

Goal State: WW_RR

Actions with associate costs:

Action-1: A person (walking or running) can move into an adjacent empty space. This move has a cost of 1.

Action-2: A person can hop over one other person into the empty space. This move has a cost of 1.

Action-3: A person can hop over two other persons into the empty space. This move has a cost of 2.

It's important to note that these actions need to be applied in order, from Action-1 to Action-3. Propose a heuristic function that can be used with the A* search algorithm to solve this puzzle efficiently. Draw the search tree that the A* search algorithm would generate to reach the goal state using your heuristic function.

Solution: