# 440/540 Artificial Intelligence HW1

**INSTRUCTIONS**

- **Due: Wednesday, 01 Feb 2023 at 23:59 PM PDT** Remember that you have 5 slip days to use at your discretion BUT you can use no more than 1 per homework.
- **Format:** Submit the answer sheet containing your answers in PDF. You can provide either typed or handwritten answers to this homework. Handwritten answers must be legible and scanned into a PDF.
- **Note:** **Please DO NOT FORGET to include your name and WSU ID in your submission.**
- **How to submit:** Submit a PDF containing your answers on Canvas
- **Policy:** See the course website for homework policies and Academic Integrity.

| | |
|---|---|
| Last Name | Patel |
| First Name | Ivani |
| WSU ID | 11809154 |

# Q1. [20 pts] Intelligent Agents

Consider a system that provides online translation of telephone conversation between English and Japanese speakers.

**(a)** [10 pts] Discuss its performance measure, environment, actuators, and sensors (PEAS)

**Answer:**
**Performance measure**: Speed, Quality, pace/tone
**Environment:** English Speaker, Japanese speakers**,** Phone Call
**Actuators:** Translator, AI base translator
**Sensors:** English Speaker's Phone, Japanese speaker's Phone

**(b)** [10 pts] Is this environment (1) fully observable (Yes/No); (2) deterministic (Yes/No); (3) static (Yes/No); (4) discrete (Yes/No); (5) episodic (Yes/No)? Do provide reasons to explain your (Yes/No) choice.

**Answer:**
(1) fully observable (Yes/No); No We are using Phone call so we cannot observe both speaker whole time and we can not observe process in phone call.
(2) deterministic (Yes/No); No We cannot no sense every part of the environment and the result of its actions on it so environment becomes inaccessible.
(3) static (Yes/No); Yes Environment rely on data-knowledge sources that don't change frequently over time.
(4) discrete (Yes/No); Yes Finite set of possibilities can drive the final outcome of the task.
(5) episodic (Yes/No)? No Translation of a sentence might depend on context of previous translation

## Q2. [15 pts] Search Nodes

Consider the graph search framework we discussed in class on a search problem with max branching factor $b$. Each search node $n$ has a backward (cumulative) cost of $g(n)$, an admissible heuristic of $h(n)$, and a depth of $d(n)$. Let $n_c$ be a minimum-cost goal node, and let $n_s$ be a shallowest goal node.

For each of the following, give an expression that characterizes the set of nodes that are explored before the search terminates. For instance, if we asked for the set of nodes with positive heuristic value, you could say: for all $n$, such that $h(n) \geq 0$. Don't worry about ties (so you won't need to worry about > versus ≥). If there are no nodes for which the expression is true, you must write "none."

**(a)** [5 pts] Give an inequality in terms of the functions defined above to describe the nodes $n$ that are explored in a breadth-first search before terminating.

**Inequality: All** ***n*****, such that:**
d(n) ≤ d(s): BFS expands all nodes which are shallower than the shallowest goal. Our search performs the goal−test after popping nodes from the fringe, so we typically expand some nodes at depth s, before we expand the optimal goal node.

**(b)** [5 pts] Give an inequality in terms of the functions defined above to describe the nodes $n$ that are explored in a best first search (or uniform cost search) before terminating.

**Inequality: All** ***n*****, such that:**
g(n) ≤ g(c): Uniform cost search expands all nodes that are closer than the closest goal node. Our search performs the goal − test after popping nodes from the fringe (ensures optimality), so we might expand some nodes of cost g(c), before we expand the optimal goal node.

**(c)** [5 pts] Give an inequality in terms of the functions defined above to describe the nodes $n$ that are explored in an A* before terminating.

**Inequality: All** ***n*****, such that:**
g(n) + h(n) ≤ g(c): All nodes with a total cost of less than g(c), get expanded before the goal node is expanded. This can be proved by induction on the cost g(n) + h(n). Considering a node n1 which satisfies this property. Its parent n0, will also satisfy this inequality, and by the induction hypothesis, n0 will be expanded before the goal is expanded, which means that it will put n1 on the fringe, which will get expanded before the goal node is expanded.

## Q3. [30 pts] Search in Graph

Consider the vacuum world problem with state space show in Figure 1 below. Let the state numbers assigned in Figure 1. Let the initial state be state 1 and the goal state be either 7 or 8.

Assume the order of in which states are added to frontier (in graph search) follow the order (S, R, L) in which actions are examined. Depending on whether we use breadth-first search (BFS) or depth-first search (DFS), the state at the front or back of the frontier will be expanded first.

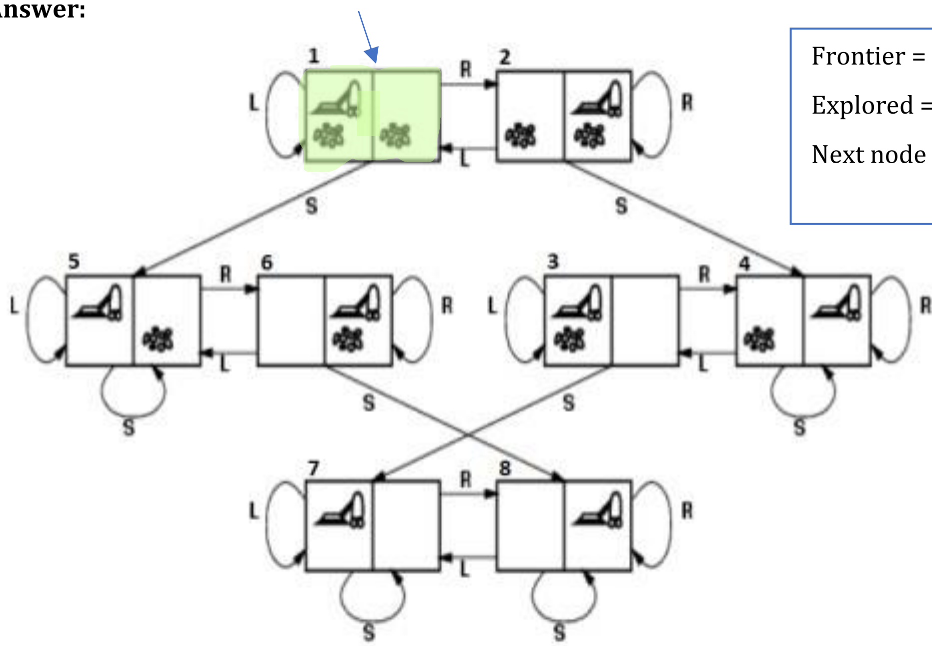
**(a)** [10 pts] Given a trace of the BFS (graph-search) algorithm in the following style: show the search tree at each stage (repeated states are eliminated) with (1) nodes in **frontier** and **explored** annotated with different color; (2) indication of next node to be explored. Assume nodes are only tested for goal or put in **explored** when it is their turn to be explored.

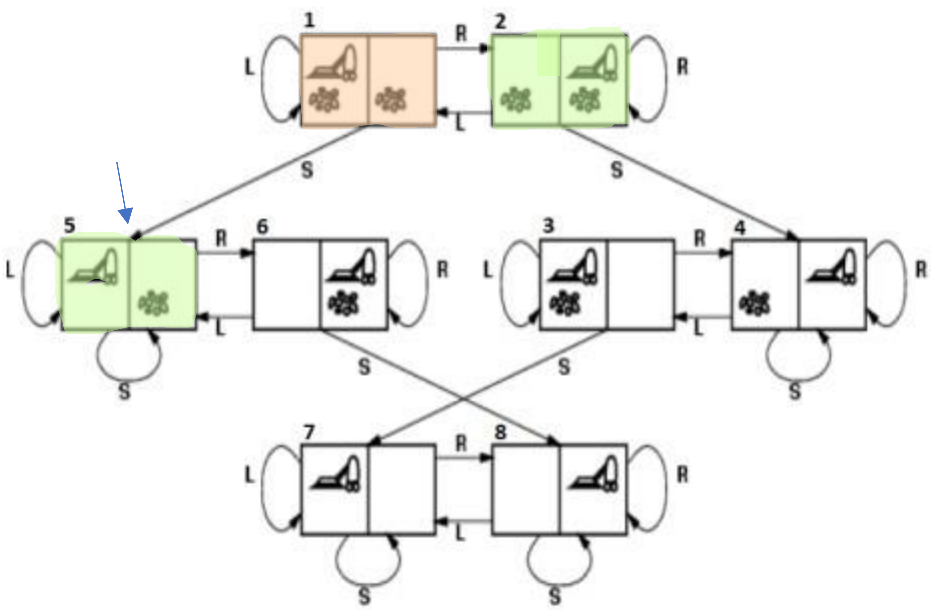
**Answer:**

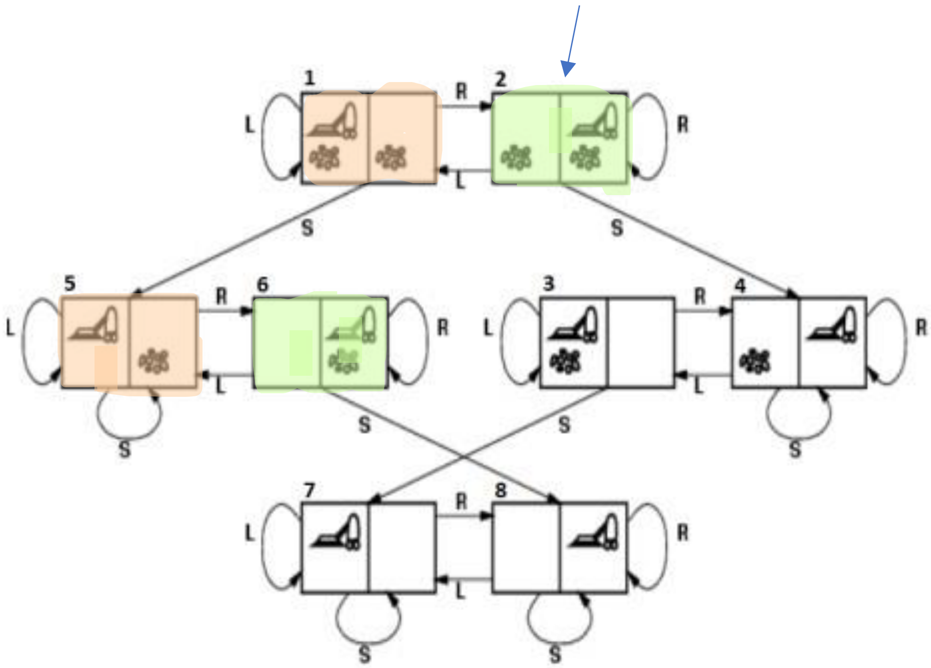
Frontier =

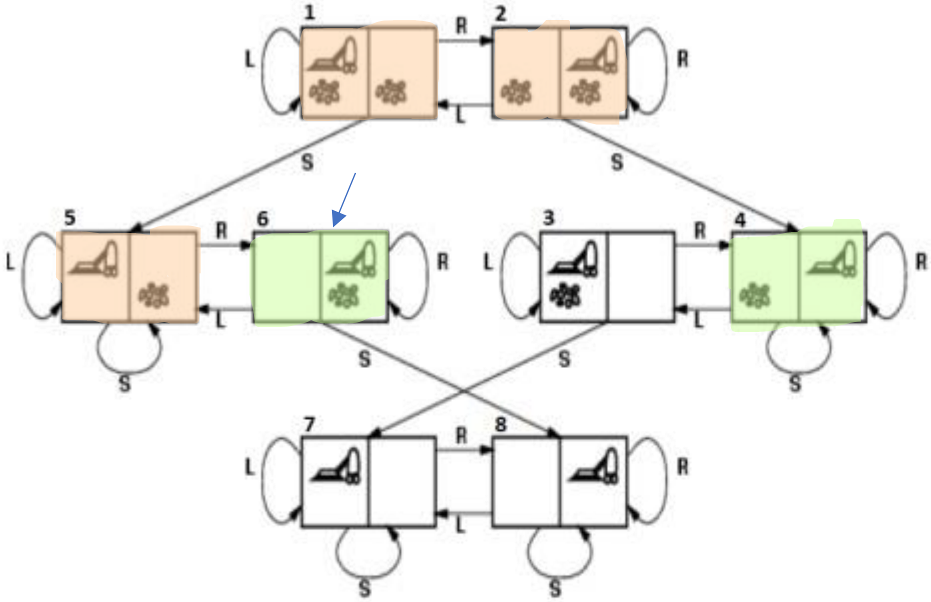
Explored =

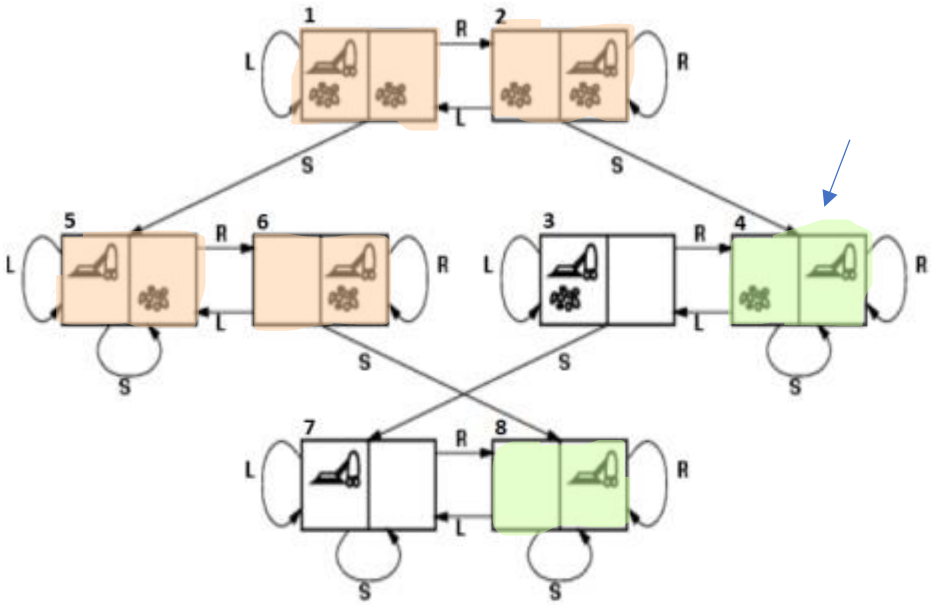
Next node to be explored = (Arrow)

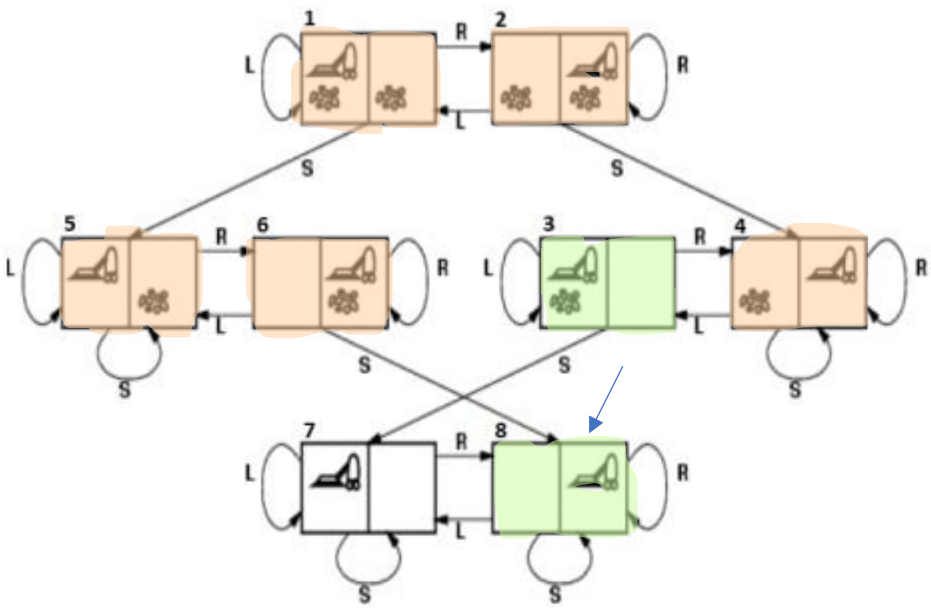


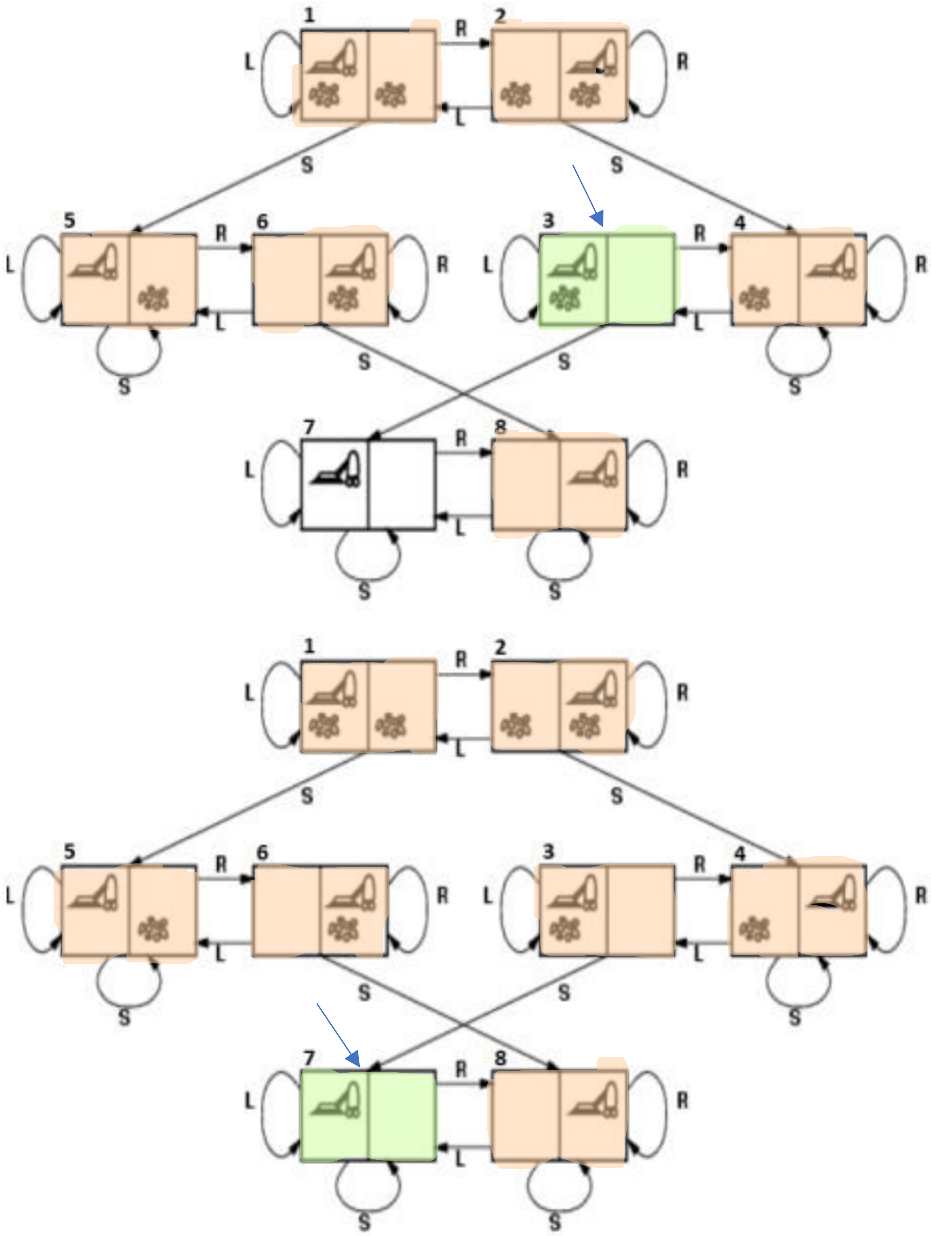




1
2
3
4
5
6
7
8
L
R
S


1
2
3
4
5
6
7
8
L
R
S


1
2
R
L
L
R
S
S
5
6
3
4
R
R
L
R
L
L
L
R
S
S
S
S
7
8
R
L
R
L
S
S


1
2
R
L
L
R
S
S
5
6
3
4
R
R
L
R
L
L
L
R
S
S
S
S
7
8
R
L
R
L
S
S


1
2
3
4
5
6
7
8
L
R
S

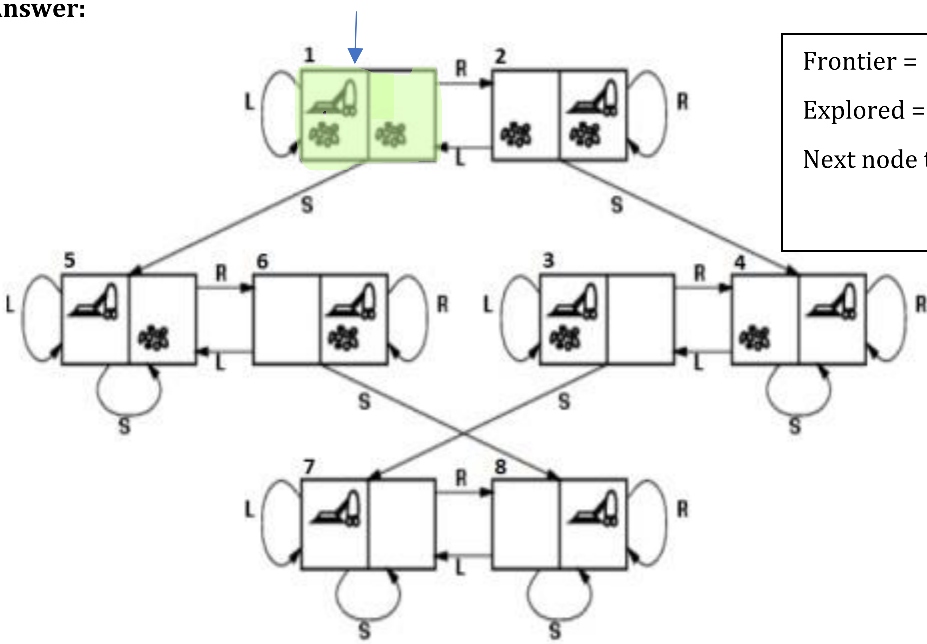
**(b)** [10 pts] Given a trace of the DFS (graph-search) algorithm in the following style: show the search tree at each stage (repeated states are eliminated) with (1) nodes in **frontier** and **explored** annotated with different color; (2) indication of next node to be explored. Assume nodes are only tested for goal or put in **explored** when it is their turn to be explored.

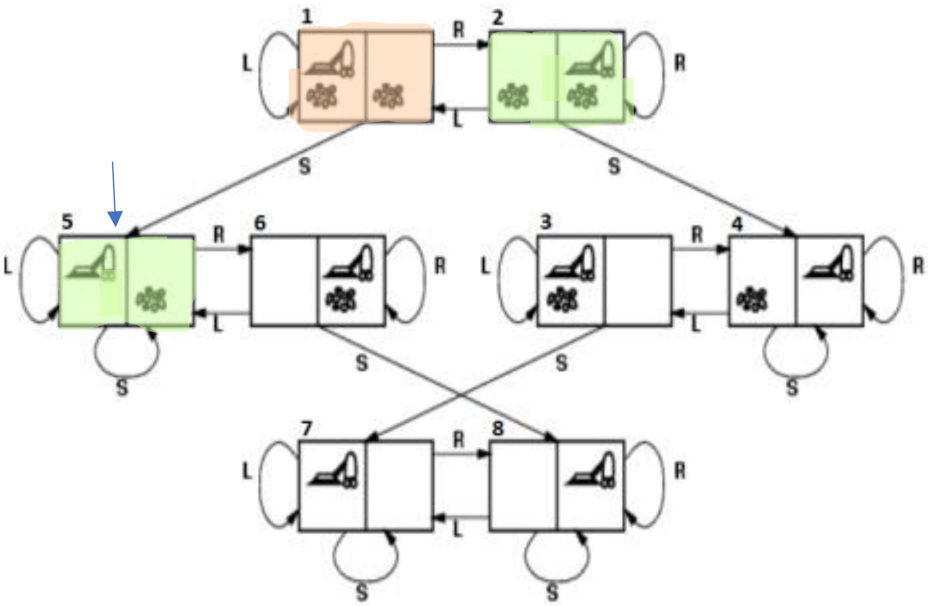
**Answer:**

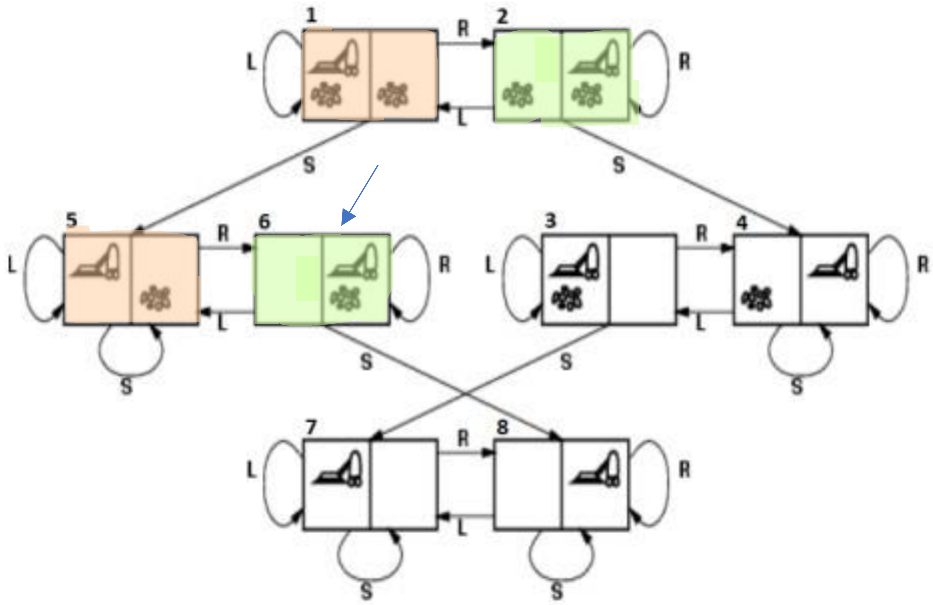
Frontier =

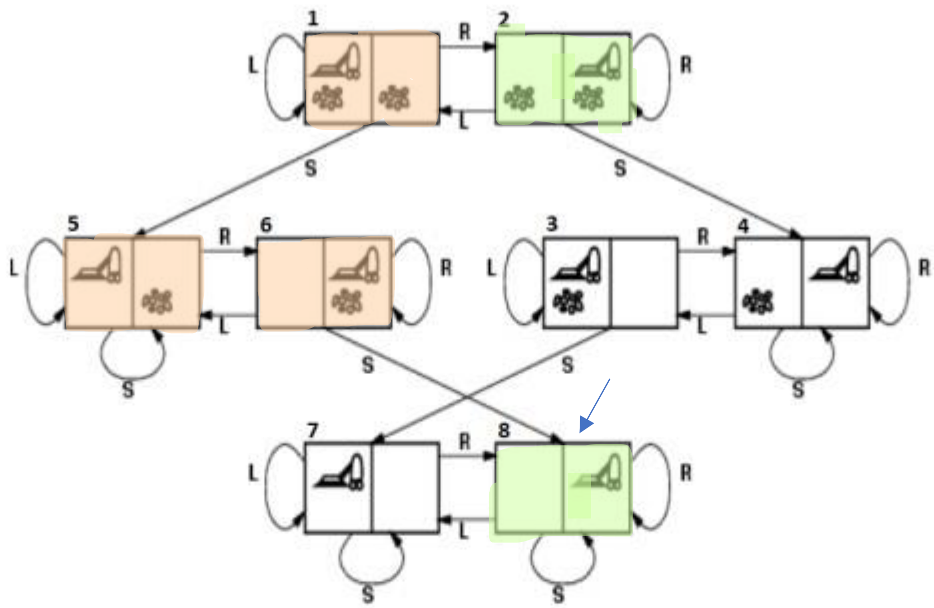
Explored =

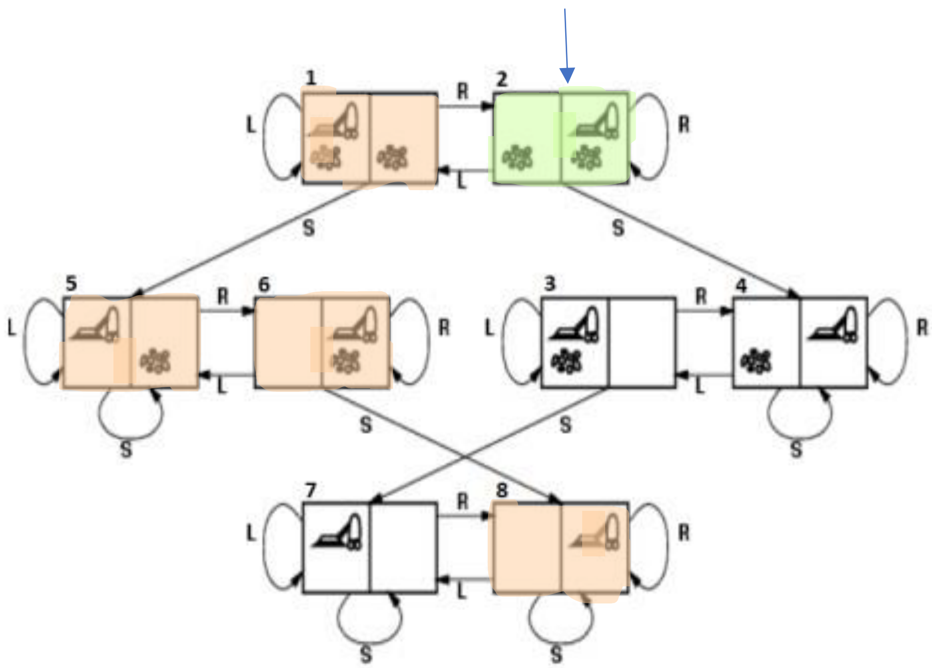
Next node to be explored = (Arrow)

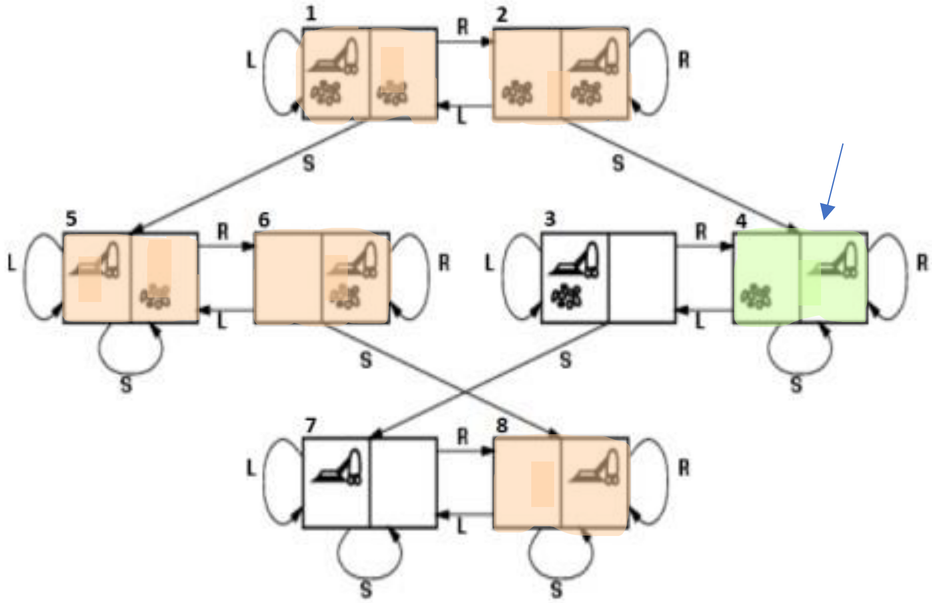


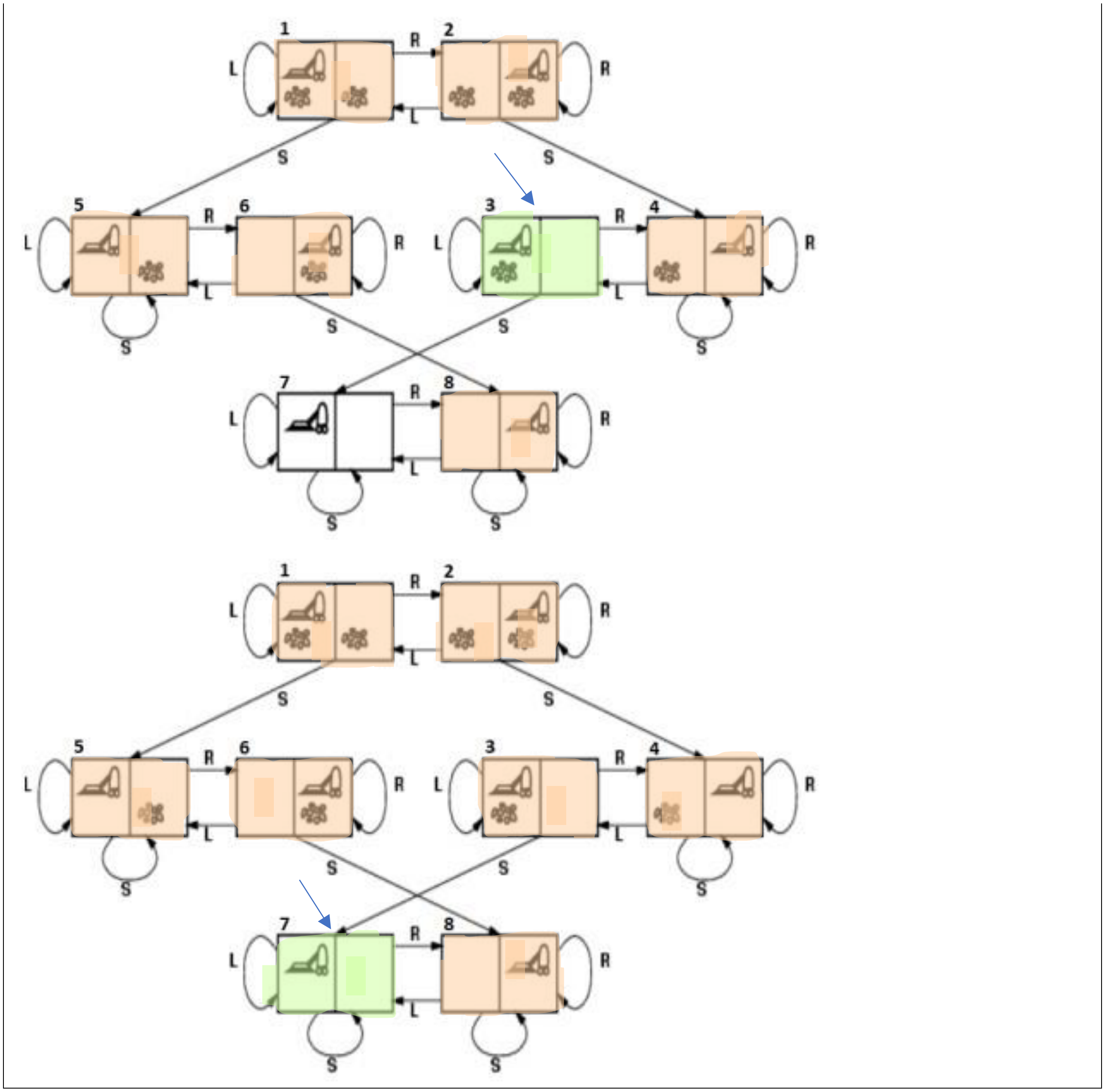



1
2
R
L
L
R
S
S
5
6
3
4
R
L
R
L
R
L
L
R
S
S
S
S
7
8
R
L
L
R
S
S


1
2
R
L
L
R
S
S
5
6
3
4
R
L
R
L
R
L
L
R
S
S
S
S
7
8
R
L
L
R
S
S


1
2
R
L
L
R
S
S
5
6
3
4
R
L
R
L
R
L
L
R
S
S
S
S
7
8
R
L
L
R
S
S


1
2
R
L
L
R
S
S
5
6
3
4
R
L
R
L
R
L
L
R
S
S
S
S
7
8
R
L
L
R
S
S


1
2
R
L
L
R
S
S
5
6
3
4
R
R
L
R
L
L
R
L
S
S
S
S
7
8
R
L
L
R
S
S
1
2
R
L
L
R
S
S
5
6
3
4
R
R
L
R
L
L
R
L
S
S
S
S
7
8
R
L
L
R
S
S

**(c)** [10 pts] Which of these two algorithms is better for this problem? Is one search strategy always better than the other in general? Do provide explanation.

**Answer:**
For this problem DFS (graph-search) algorithm is better as it consumes less memory than the BFS: once a node has been expanded, it can be removed from memory as soon as all its descendants have been fully explored. But the algorithm may perform long searches when the solution is simple (when the goal is close to the root vertex).
BFS is better when target is closer to Source. So, when we search for the vertices that stay closer to any given source BFS works better. DFS is better when target is far from source. So, when we can find the solutions away from any given source DFS works better. DFS is more suitable for decision tree.

# Q4. [20 pts] Adversarial Search

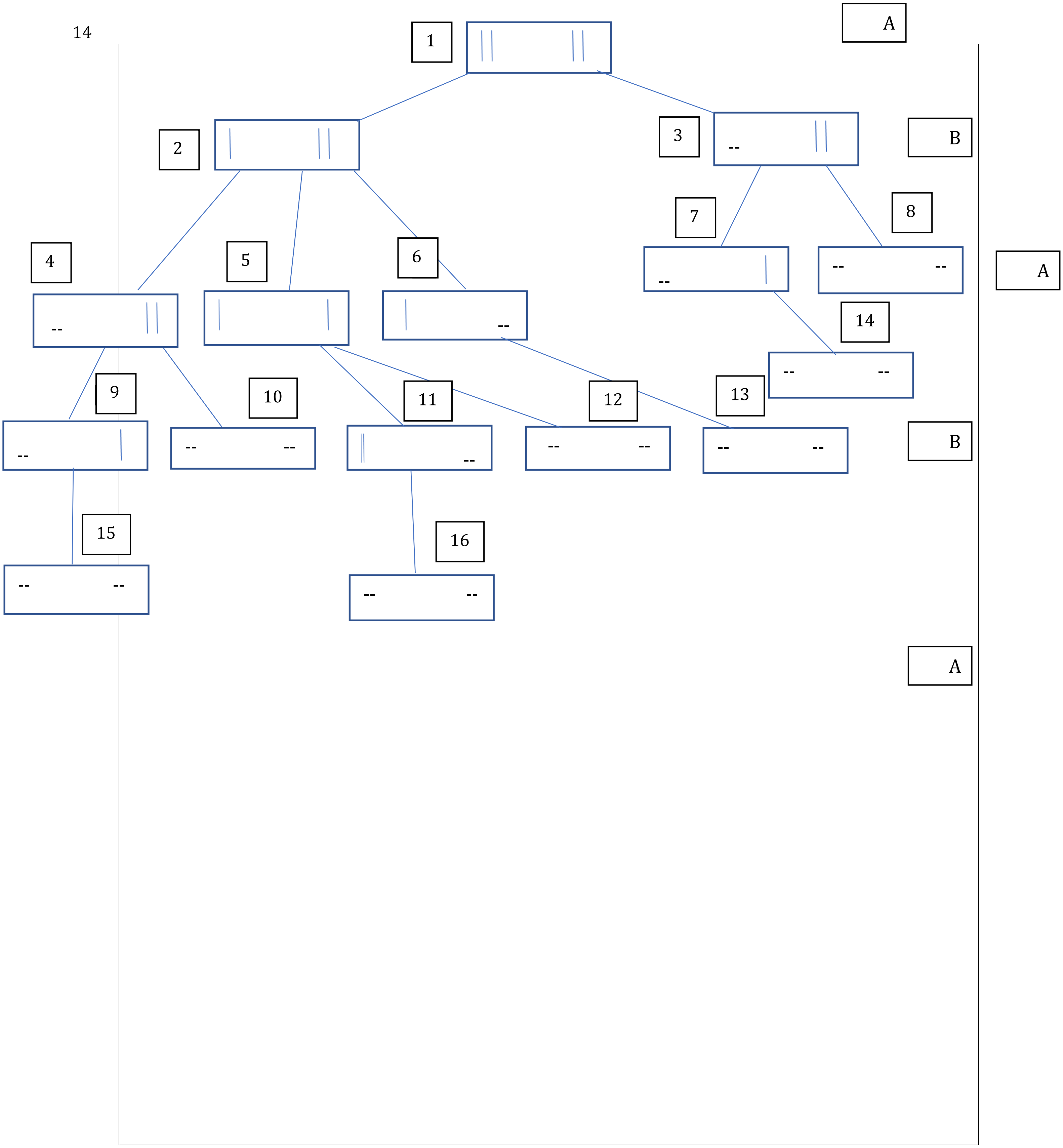
Consider the following game between two players: (1) there are two piles, each with two sticks; (2) each player might take either one or two sticks from an existing pile during his/her turn; (3) the player who picks the last stick wins.

**(a)** [5 pts] Propose a representation for the state of this game.

**Answer:**
1. There are two players.
2. Initial state: There 2 piles of sticks, 2 sticks in each pile.
3. Moves are made by alternate players.
4. Admissible transformations: each player might take either one or two sticks from an existing pile during his/her turn.
5. Terminal state: When both piles have no sticks, there are no more moves and the game is over.
Outcome: Whoever takes the last stick wins.

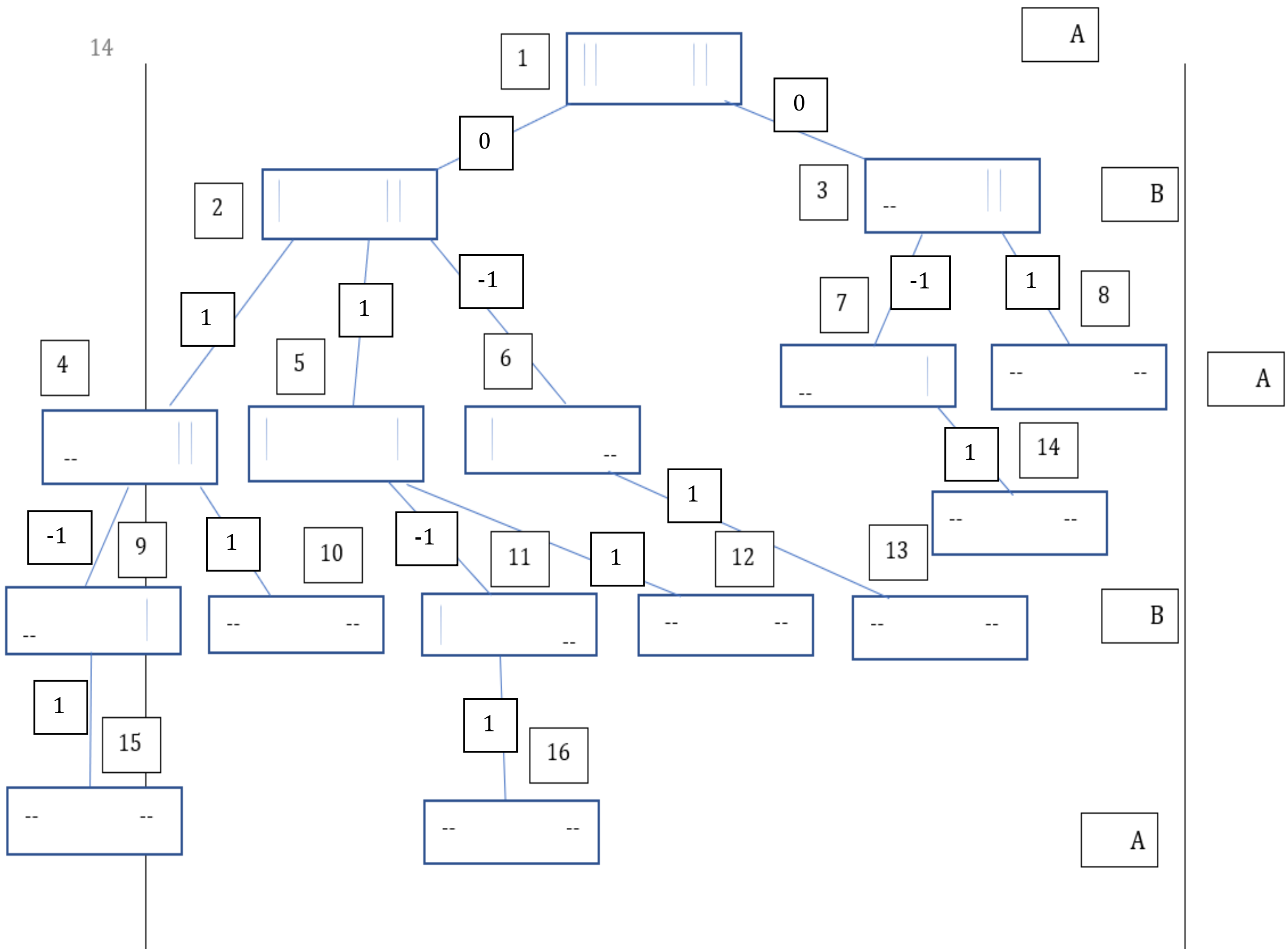
**(b)** [5 pts] Using your proposed presentation, draw the game tree for this game.

**Answer:** The boxes represent each state. The boxes occur at each point at which a player has to make an admissible transformation. The interior of the boxes represents the piles of matches. The upper-case letter by the box indicates the player


A
1
2
3
B
7
8
4
5
6
A
14
9
10
11
12
13
B
15
16
A

**(c)** [5 pts] Let a state that results in a win for the first player (MAX) be of value 1 while a state that results in a win the the second player (MIN) be of value -1. Solve the game by assigning a value for each node in the game tree above. If you want to win, would you opt to move first or second? Do provide an explanation.

**Answer:**



At states 8,15 and 16, B wins. At states 10, 12, 13 and 14, A wins. At state 15 B wins, so at state 9 A has already lost, so at state 4 B will choose to go to 9, so state 4 is already a winning position for B. With this type of analysis, we discover B has lost the game before it starts, for a winning strategy for A is : If at state 2 go to sate 6, if at state 3 go to state 7. So I would opt to move first.

**(d)** [5 pts] Suppose instead of two piles of two sticks each, we have two piles of three sticks each. Now, would you opt to move first or second? Do provide an explanation.

**Answer:**
I would opt to move second as who ever picks first stick will end up getting less chance of making game in his favor. If first one picks one stick and second one takes either one or two stick, he ends up winning the game.
If first player takes out two sticks, and second player either takes one or two stick he again ends up winning the game.
-For example, we have 2 piles of 3 sticks. Suppose player A take one stick and player B take 2 sticks then player A either take 1 stick or 2 sticks player B wins. Now if player A takes one stick and player B takes one stick then again player A either takes one stick or two sticks player B wins as he becomes the one taking the last stick.
-Now suppose player A takes 2 sticks and player B takes 2 sticks then in next weather player A takes either 1 stick or two sticks player 2 wins . Now if player A takes 2 sticks first and player 2 takes 1 stick then player 1 either takes 1 stick or 2 sticks player 2 wins because again he becomes the one taking the last stick.
-So second player wins in either of the condition.

# Q5. [15 pts] Heuristic-Guided Search

Consider the 8-puzzle that we discussed in class. We know that there are two admissible heuristic functions: (1) $h_1(n)$ denotes the number of misplaced tiles; and (2) $h_2(n)$ denotes the sum of distances of the tiles from their goal position. Suppose now we define two new heuristic functions: $h_3(n) = (h_1(n)+h_2(n))/2$ and $h_4(n) = h_1(n)+h_2(n)$.

**(a)** [10 pts] Is $h_3$ and $h_4$ admissible? Do provide an explanation.

**Answer:** Since $h_1(n) \leq h_2(n)$,
$h_3(n) = (h_1(n)+h_2(n))/2 \leq (h_2(n)+h_2(n))/2 = h_2(n) \leq h*$
hence h3 is admissible.
On the other hand, h4 is not admissible. Considering a board in which moving one tile will reach the goal. In this case, $h_1(n) = h_2(n) = h* = 1$, and
$h_4(n) = h_1(n) + h_2(n) = 1+1 > h*$

**(b)** [5 pts] If admissible, compare their dominance with respect to $h_1$ and $h_2$.

**Answer:**
Since $h_1(n) = (h_1(n)+h_1(n))/2 \leq (h_1(n)+h_2(n))/2 = h_3(n)$ we have $h_1(n) \leq h_3(n) \leq h_2(n)$. That is, $h_2(n)$ dominates $h_3(n)$ and $h_3(n)$ dominates $h_1(n)$.

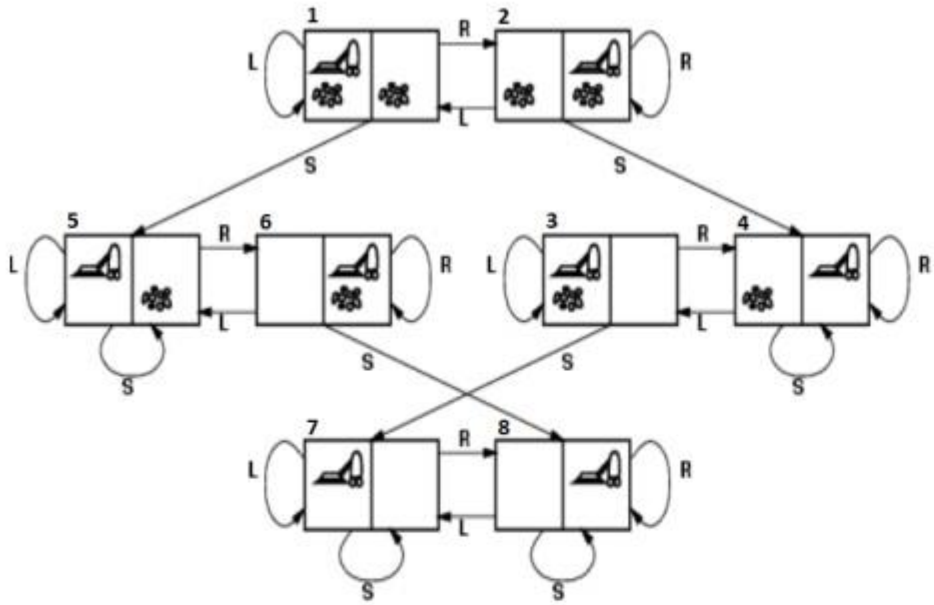



Figure 1: The state space of the vacuum world.