

Homework #1: Chapters 1, 2, 3

1.

Agent Type	Performance Measure (Quant)	Environment (Env/Agent=Ext/Int)	Actuators	Sensors
(a)	% of games won	tic tac toe matrix, opponent	function to put an X in an open square	function to read state of the board
(b)	% of correct answers, % of questions answered	web pages, question strings	display answer to question, internet connection to request a web page (HTTP GET)	question input, internet connection to read requested web pages
(c)	% of fruit/vegetables correctly identified	fruit, vegetables	screen to output identification on	camera(s), scale

2.

Dimension	Agent		
	(a) Tic-Tac-Toe	(b) Query Answer System	(c) Fruit/Veggy Scanner
observable	fully (the entire board is visible at all times)	partially (the agent can see at most one web page at a time)	fully (if you assume multiple cameras to see from all angles, perfect lighting, or that state consists of a single image) / partially (if some information is obscured)
deterministic	strategic (next state is fully determined by agent's action and its opponent's action)	stochastic (the agent has no control over the web pages, server may be down)	stochastic (the agent has no control over the next item produced)
episodic	sequential (each move affects possibility of winning in the future)	sequential (the page sensed is the page requested during the last state, agent may need to follow links on some web pages previously requested)	episodic (each new fruit or vegetable represents a new episode)
static	static (board can't change until agent has made its move)	dynamic (the Web is constantly changing)	dynamic (e.g., dust storms, weather conditions)
discrete	discrete (the board is finite, and there are a finite # of moves)	continuous (the number of combinations of words in the question and on the pages is essentially infinite)	continuous (camera images are digital, but the large # of possible image makes case-wise analysis impossible)
# of agents	multi (there is an opponent)	single (other web users and web pages authors do not directly affect the agent, can be treated as part of the environment)	single (once the item is on the scanner, the customer or clerk is irrelevant)

3. If the environment is **fully observable**, then the table-driven agent can be rational without considering percept history. When the environment is only **partially observable**, then it is better to use the entire percept history.

Alternatively, if the environment is **episodic** then the table-driven agent can be rational without considering percept history. However, a **sequential** environment does not necessarily require percept history for rationality; in fact it is only needed if the environment is also **partially observable**.

4. Initial State: (5 points)

Some configuration consisting of a set of areas N , an adjacency function $\text{adjacent}(x) = \{y \mid \text{such that } y \text{ is adjacent to } x\}$ and a coloring function $\text{color}: N \rightarrow \{1,2,3,4,\text{uncolored}\}$.

Successor Function: (10 points)

for a state s , the $\text{SUCCESSIONS}(s)$ are generated as follows:

- for each $x \in N$ such that $\text{color}(x) = \text{uncolored}$
 - for each $c \in \{1,2,3,4\}$ such that there does not exist a $y \in \text{adjacent}(x)$ where $\text{color}(y) = c$
 - $\{\text{set } x\text{'s color to } c, s'\} \in \text{SUCCESSIONS}(s)$ where s' is s with $\text{color}(x) = c$

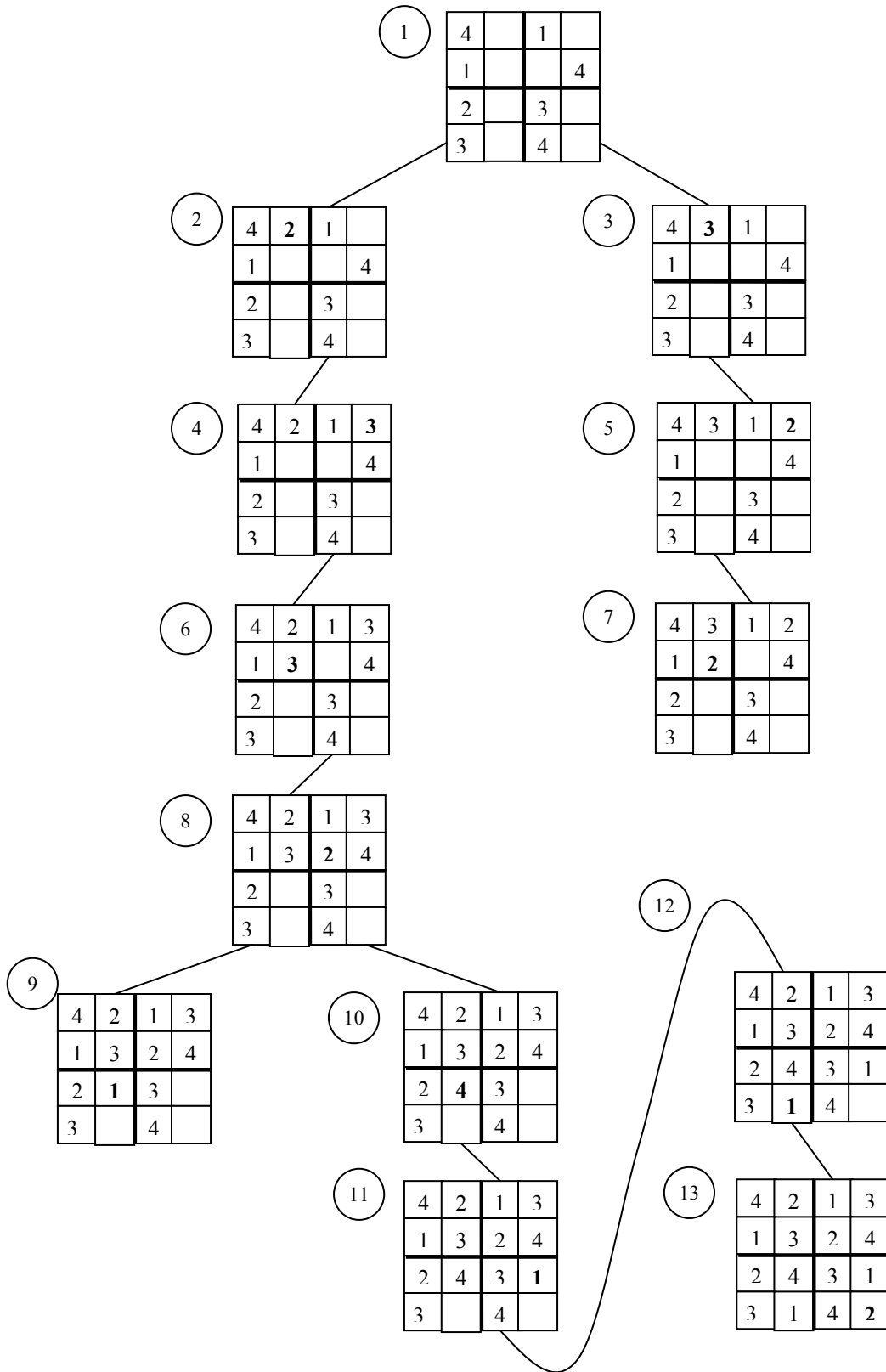
Goal Test: (3 points)

true if there is no $x \in N$ such that $\text{color}(x) = \text{uncolored}$.

Path Cost Function: (2 points)

All that matters for this problem is that a solution is found. No solution is considered any better than any other. Thus path cost could be 0 or 1. A path cost calculated on 1 per step is also acceptable, since all solutions require the same number of steps (i.e., $|N|$).

5.



6.

