CPS 570: Artificial Intelligence
Practice final exam

Your name:

Please read instructions carefully, and circle your final answers where appropriate. Do not worry if you cannot finish everything. Do not write down disorganized answers in the hope of getting partial credit; it's better to do a few questions completely right. You can use extra pages. Good luck!

–Vince

Question 1: True or false.

Label each of the following statements as true or false. You do not need to give any explanation.

1. Problems that are easy for humans to solve are also easy for AI.
2. In constraint satisfaction/optimization, the runtime is not affected by the way in which we choose the next variable to set (as long as we set only one variable at a time).
3. If we use alpha-beta instead of the standard minimax algorithm (in a two-player zero-sum perfect-information game such as chess), but we look all the way to the bottom of the tree (no limited-depth cutoff), then we are guaranteed to still get an optimal (that is, minimax) solution for the game.
4. If we use the standard minimax algorithm (in a two-player zero-sum perfect-information game such as chess), but we look only to a limited depth and apply an evaluation function there, then we are guaranteed to still get an optimal (that is, minimax) solution for the game.
5. For any setting where we have a finite number of variables that each can take only finitely many values, we can represent any probability distribution using a Bayes net.
6. Given a Bayes net and a binary variable X in it, we know how to compute $P(X = \text{true})$ in polynomial time.
7. If we transform a POMDP to an MDP (by using the belief-state trick), then the number of states does not change.
8. In game theory, the following can happen: each player plays her dominant strategy, and still, both players are unhappy, in the sense that there is another outcome of the game that makes both players happier.

Question 2: Using A* to solve a juggling-like problem.

We have a robot with *three* hands, Left, Middle, and Right, which can each hold one ball. There is also a table on which we can put one ball. Currently, the Left hand is holding a Red ball, the Middle hand is holding a Green ball, the Right hand is holding a Blue ball, and the table is empty. We represent this state as $(R, G, B, \cdot)$. Our goal state is $(B, R, G, \cdot)$ (that is, we have rotated the balls, and the table is empty). There are three types of action:

- Place one of the balls on the table (if the table is empty).
- Pick up a ball from the table, using an empty hand.
- Transfer a ball from one hand to an empty hand.

For example, we could first place the Middle hand's ball on the table, resulting in $(R, \cdot, B, G)$; then, we could transfer the Right hand's ball to the Middle hand, resulting in $(R, B, \cdot, G)$.

The cost of a solution is the number of moves. Use A* (with forward search, i.e., not backwards or bidirectional) to find an optimal solution. As an admissible heuristic, you should use: the number of balls that are not currently in the correct location (at the beginning, this evaluates to 3). Please draw the entire search tree, including the g and h values. Hint: if the $g + h$ value of two nodes is the same, break ties towards nodes that are deeper in the tree (this will get you to a solution faster, you only need to find one optimal solution).

Please note that, for this question, you do *not* need to represent the actions in a standard planning language (like
MoveToTable(ball1, hand1)
Precondition: IsHolding(hand1, ball1)
....)
—though, of course, you could.

Question 3: Planning to brush our teeth.
In this question, we are going to create a plan to brush teeth.

Start: is_brush(Toothbrush$_0$) is_paste(Toothpaste$_0$) is_tap(Tap$_0$)
Brush(b,x) *Preconditions:* is_brush(b) is_wet(b) has_paste(b) *Effects:* is_foamy(b) is_foamy(x) is_clean(x)
Hold_Under(b,t) *Preconditions:* is_tap(t) is_running(t) *Effects:* is_wet(b) ¬has_paste(b) ¬is_foamy(b)
Goal: is_clean(Mouth$_0$) ¬is_foamy(Mouth$_0$) ¬is_foamy(Toothbrush$_0$) ¬is_running(Tap$_0$)

Note that taps are quite powerful: they will blast any toothpaste off the brush. Also note that this is a slightly different language from the STRIPS language: negative literals can be part of the state, unmentioned literals are assumed unknown.

a Create three more actions, **Turn_on**, **Turn_off**, and **Apply_paste**, to make the problem feasible.

b Create a partial-order plan for this problem. Point out where there is flexibility in how the actions are ordered.

c Presumably, your plan does not turn off the water while you are brushing, that is, you are wasting water. Can you somehow add "not wasting water" as a goal, so that your plan will turn off the water while you are brushing (and then turn it back on for rinsing)? Discuss the modifications you need to make, and whether this is a good/elegant way of addressing this issue.

Question 4: Does our friend have a job?

Our friend John does not have a stable job. He manages to find work sometimes, but he is often unemployed as well. He does not like to talk about whether he has a job. However, we can infer something about whether he has a job, based on whether he visits us (he is more likely to visit us if he does not have a job).

We consider a three-day model with $t \in \{\text{Wednesday, Thursday, Friday}\}$. J_t is true if John has a job on day t. V_t is true if he visits us on day t.

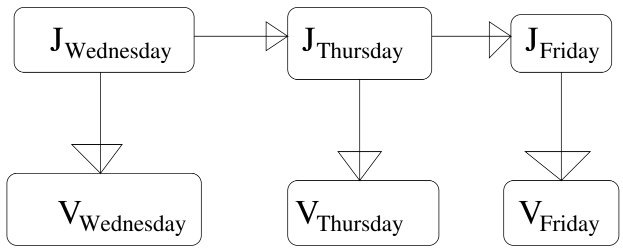

Figure 1: A Bayes net for whether John has a job.

We have:

$P(J_{\text{Wednesday}} = \text{true}) = .5$

$P(J_{t+1} = \text{true}|J_t = \text{true}) = .8$ for $t \in \{\text{Wednesday, Thursday}\}$
$P(J_{t+1} = \text{true}|J_t = \text{false}) = .3$ for $t \in \{\text{Wednesday, Thursday}\}$

$P(V_t = \text{true}|J_t = \text{true}) = .2$ for $t \in \{\text{Wednesday, Thursday}\}$
$P(V_t = \text{true}|J_t = \text{false}) = .6$ for $t \in \{\text{Wednesday, Thursday}\}$

$P(V_t = \text{true}|J_t = \text{true}) = .5$ for $t \in \{\text{Friday}\}$
$P(V_t = \text{true}|J_t = \text{false}) = .9$ for $t \in \{\text{Friday}\}$

Suppose John visits us on Thursday, but not Wednesday or Friday. What is the probability that John had a job on Friday, that is, what is $P(J_{\text{Friday}}|V_{\text{Wednesday}} = \text{false}, V_{\text{Thursday}} = \text{true}, V_{\text{Friday}} = \text{false})$?

(Hint: compute quantities such as $P(J_{\text{Thursday}} = \text{false}, V_{\text{Wednesday}} = \text{false}, V_{\text{Thursday}} = \text{true})$ and use Bayes' rule. Note that this is a typical HMM problem, except the observation model is not stationary: visits are more likely on Friday than on other days.)

Question 5: A Markov decision process for sports.

In this question, you will solve a simple Markov decision problem by hand. This question involves some sport like soccer or basketball. We control one of the two teams. When you have the ball, you can Shoot (try to score) or Pass (try to give the ball to a teammate). There are four states: Far (we have the ball far away from the opponent's goal/basket), Close (we have the ball close to the opponent's goal/basket), Lost (we just lost the ball), Scored (we just scored a goal/basket). In the Lost or Scored states, we cannot take any action, but we will automatically transition to the Far state in the next period. (Equivalently, you can imagine that you can take one of the actions in those states, but it will not affect anything.) Note that we never play defense, we just try to score as many goals/baskets as possible. The following figure gives the transition probabilities and the rewards. We only get a reward in the Scored state, of 1;[1] everywhere else rewards are 0.

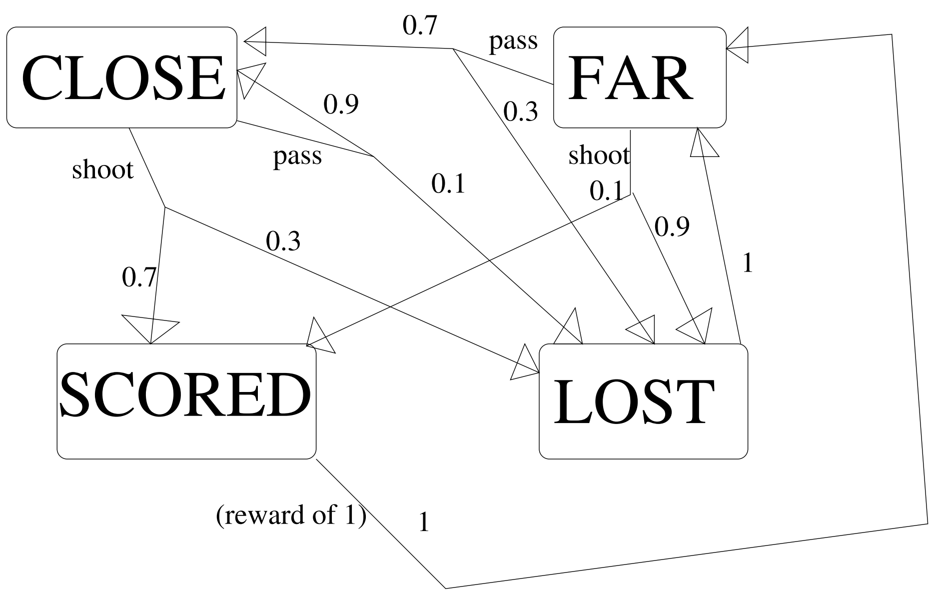

Figure 2: Sports MDP.

The discount factor is $\delta = .9$. Solve this MDP, using any technique that you like. You can probably guess the optimal policy quite easily, but you should also give the value of being in each state (if you cannot give it exactly, you should give it at least approximately).

[1]Baskets scored from far away do not get extra points in this problem...