1. This is an open-book, open-notes exam, but NOT open-neighbor. Your work must be your own!
2. You will work on your laptop or desktop computer and submit your answers in the form of a text file whose name is formatted thus: 
   <your last name>_<initial>_mt.txt. Email the file to me at the email address appearing in the wiki. If you have questions about the submission, contact the instructor as soon as possible.
3. You have 2 hours for the exam.
4. The exam is out of 100 points.

1. (2 points) T/F In a game using Markov Decision Process, the value of a state is the average of the values of its successor states.

2. (2 points) We know that the each successor state of a state in an MDP game tree (for example, a robot car driving simulation game such as the one we covered in class) is associated with a probability value. State one way in which these probabilities can be obtained in a real-life situation, for example, if you were trying to write a robot car driving simulation game for the market.

Questions [3-8] refer to the Bellman equation for computing the optimal value of a state in an MDP game, given below:

\[ V^*(s) = \max_a \sum_{s'} T(s, a, s') \left[ R(s, a, s') + \gamma V^*(s') \right] \]

Consider this MDP game: A program simulating a stock market speculation game assumes that you have $100 to start with, in a brokerage account which charges you 1% of the day’s balance as fees. On January 2, 2014, you get a tip that IBM stock is about to go up exactly 10% the next day. Another tipster tells you that it will remain exactly the same. You have no other information and you trust both sources equally.

You r choice is to invest the entire $100 in IBM for a day, or to leave the cash in the account.

3. (2x7= 14 points) Which term in the Bellman equation best represents each of the following? (equation symbols only, not numbers!)
   a. The 1% interest
   b. Your balance on January 2, 2014 at the opening of the market?
c. Your balance on January 3, 2014 at the opening of the market?
d. Your order to the brokerage regarding investment in IBM?
e. The probability that your tip regarding the IBM stock rise may or may not be accurate?
f. Your degree of trust in your tipsters
g. The profit you might make on a day’s transaction

4. **(8 points)** Suppose (for some odd reason) you want to select the action that results in the median value of the possible outcome states, instead of the maximum value. Rewrite the Bellman equation to show this modified criterion. Do you think it is still appropriate to use the $V^*$ notation? Explain why or why not.

5. **(4 points)** Let $V_0(s), V_1(s), ...$ represent the values of your balance at the start of January 2, January 3, and so on, with $s$ representing the states of the game. How many possible states are there and what are they?

6. **(9 points)** Using the states you identified in problem 5 above, show the value vectors for $V_0$, $V_1$, and $V_2$ as a table.

7. **(8 points)** Will this game converge? Explain why or why not.

8. **(7 points)** What parameter of the game would you change, to make it converge and to what value would you change it? Explain your answer.

9. **(2 points)** T/F Utility of a lottery is the expected value of the utilities of each choice in the lottery.

10. **(2 points)** T/F Utility of betting on a horse race is the expected dollar value of the payoff for each horse in the race.

11. **(2 points)** T/F The Bellman equation has no relationship to the maximum expected utility principle.

**Slider puzzle:** The following questions (12-16) are based on the slider puzzle. A slider puzzle contains several small squares that fit into a rectangle with a hole in it. For example:

```
1 0 0
0 0
```

A move consists of sliding a piece into the empty square. Thus, two moves are possible from the example above:
The goal of the slider puzzle is to reach a specified target state from an initial state.

You can solve a slider puzzle by doing a search on it, where every step of the search is sliding one piece to a new place.

If a search uses a **heuristic estimate** of the distance to the goal, take that heuristic distance metric to be the number of pieces that are in a place that should be blank or occupied by a piece of the other type. (in the wrong place, in other words) For example, if the goal is:

```
1 0 0
0 0 0
```

then the **heuristic value** associated with the following configuration is 2 because 0 is where 1 should be and the 1 is where a blank should be.

```
0 1 0
0 0 0
```

The actual distance between two states is the number of moves taken to get from the first state to the second.

Break ties according to the position of the blank space when you scan left to right in the top row, then the bottom row as in the following examples:

```
1 0 0
0 0 0  
```

comes before

```
1 0 0
0 0 0
```

which comes before

```
1 0 0
0 0 0
```

When showing your solution in text form, use the following notation for the states:
[P(1,1), P(1,2), P(1,3), P(2,1), P(2,2), P(2,3)] where P(i,j) is the contents of the square at the i-th row, j-th column, taking the top left square as (1,1). The contents can be 1, 0, or B for blank. For example [1,B,0,0,0,0].

When showing a tree, use the following notation:
Parent -> Child 1, Child 2, Child 3, ... with Child 1, Child 2, Child 3 being the children from left to right. Mark levels of a tree as Level I, If there are scores or metrics or other numbers associated with a node, show them in angle brackets.

For example,

Level 0 (root level):
[0,1,B,0,0,0] <2> → [1,B,0,0,0,0]<1>, [1,0,B,0,0,0]<3>, [1,0,0,0,0,B]<2>

Level 1:
[1,B,0,0,0,0]<1> → [B,1,0,0,0,0]<4>, [1,0,B,0,0,0]<3>, [1,0,0,0,B,0,0]<2>

and so on.

12. (8 points) Breadth-first search:

Do a breadth-first search from 

![Start State](image1)

to 

![Finish State](image2)

. Show the states from start to finish state.

13. (8 points) Hill climbing: Do a hill-climbing search from 

![Start State](image3)

to 

![Finish State](image4)

i.e., Show the states you go through from the start state to the finish state using hill climbing. Include in your tree all nodes at the ends of the paths that you put on the search queue. Use the heuristic distance metric provided above.

14. (8 points) Depth first search:

If you were doing a depth-first search, would you need to allow backtracking to ensure that you find a solution for the initial and goal states shown in Q3? There is no need to show the tree. State True or False and explain why.

15. (8 points) Is the heuristic given for the slider puzzle admissible? State True or False and Explain why.

16. (8 points) A* Search. Do an A* search from
to find a solution to the puzzle that has the minimum number of moves. Show the tree. Use the heuristic distance metric provided in the introduction.