

CS221 Practice Midterm #1

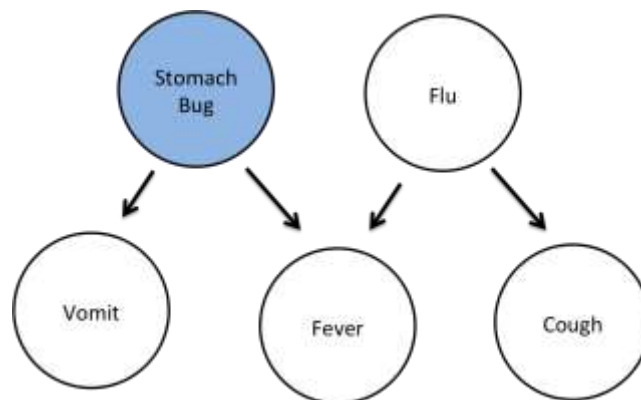
Summer 2013

The following pages are excerpts from similar classes' midterms. The content is similar to our midterm – but I have opted to give you a document with more problems rather than one that follows the structure of the midterm precisely. See the midterm handout for more details on what the exam will look like. The midterm is 2 hours. It is open book and open computer but **closed** Internet.

1. [Short Answers] Truth Test (10 points)

For the following questions, a correct answer is worth 2 points, no answer is worth 1 point, and an incorrect answer is worth 0 points. Circle true or false to indicate your answer.

- (true or false) If $g(s)$ and $h(s)$ are two admissible A^* heuristics, then their average $f(s) = \frac{1}{2} g(s) + \frac{1}{2} h(s)$ must also be admissible.
- (true or false) For a search problem, the path returned by uniform cost search may change if we add a positive constant C to every step cost.
- (true or false) The running-time of an efficient solver for tree-structured constraint satisfaction problems is linear in the number of variables.
- (true or false) The amount of memory required to run minimax with alpha-beta pruning is $O(bd)$ for branching factor b and depth limit d .
- (true or false) For a discrete bayesian network with n variables, the amount of space required to store the “joint” distribution table is $O(n)$.
- (true or false) In a markov decision problem, there are no actions and instead everything is controlled by “chance.”
- (true or false) In a markov decision problem, there are no actions and instead everything is controlled by “chance.”
- (true or false) Assume the Bayes Net from class is a perfect representation of the world:



You know that you *do not* have a stomach bug. If you were to vomit, would that information change the probability that you had a fever?

2. [Deterministic Search] Mr. and Ms. Pacman (20 points)

Pacman and Ms. Pacman are lost in an $N \times N$ maze and would like to meet; they don't care where. In each time step, both simultaneously move in one of the following directions: {NORTH, SOUTH, EAST, WEST, STOP}. They do not alternate turns. You must devise a plan which positions them together, somewhere, in as few time steps as possible. Passing each other does not count as meeting; they must occupy the same square at the same time.

- a. Formally state this problem as a single-agent state-space search problem.

States:

Goal test:

Legal actions (given a state):

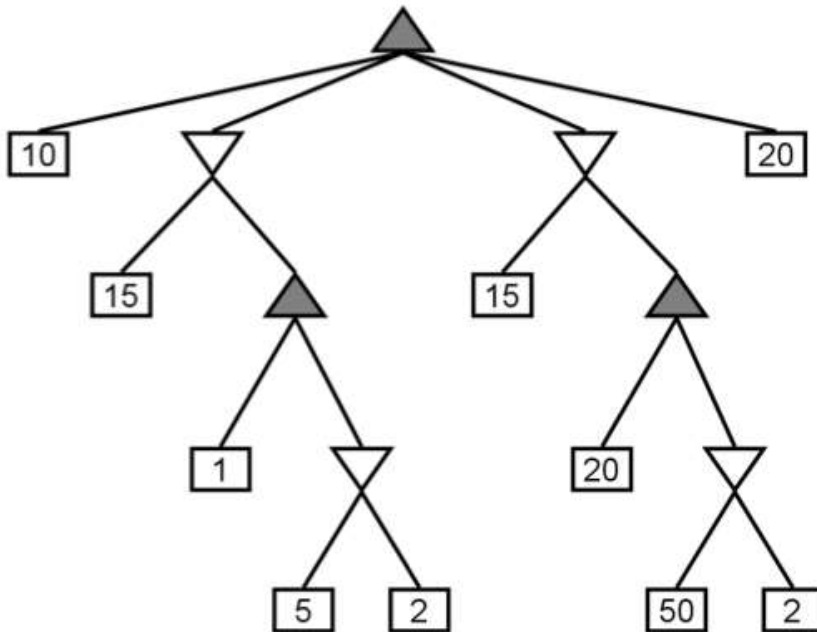
Successor function (given a state and an action):

[continued on the next page]

- b. Give a non-trivial admissible heuristic for this problem.
- c. Circle all of the following graph search methods which are guaranteed to output optimal solutions to this problem:
- (i) DFS (ii) BFS (iii) UCS
 - (iv) A* (with a consistent and admissible heuristic)
 - (v) A* (with heuristic that returns zero for each state)
- d. If h_1 and h_2 are admissible, which of the following are also guaranteed to be admissible? Circle all that apply:
- (i) $h_1 + h_2$
 - (ii) $h_1 * h_2$
 - (iii) $\max(h_1, h_2)$
 - (iv) $\min(h_1, h_2)$
 - (v) $(\alpha)h_1 + (1-\alpha)h_2$ for any value α between 0 and 1

3. [Adversarial Search] MiniMax (15 points)

Consider the following minimax tree:



- What is the minimax value for the root?
- Draw an X through any nodes which will not be visited by alpha-beta pruning, assuming children are visited in left-to-right order.
- Is there another ordering for the children of the root for which more pruning would result? If so, state the order.

[continued on the next page]

- d. Propose a general, practical method for ordering children of nodes which will tend to increase the opportunities for pruning. You should be concise, but clearly state both what to do about min nodes and max nodes.

4. [Bayes Net] Nuclear Power Station (15 points)

In your local nuclear power station, there is an alarm that senses when a temperature gauge exceeds a given threshold. The gauge measures the temperature of the core. Consider the Boolean variables \mathbf{A} (alarm sounds), \mathbf{F}_A (alarm is faulty), and \mathbf{F}_G (gauge is faulty) and the multivalued, discrete nodes \mathbf{G} (gauge reading) and \mathbf{T} (actual core temperature).

- a. Draw a Bayesian network for this domain, given that the gauge is more likely to fail when the core temperature gets too high.

- b. Suppose there are just two possible actual and measured temperatures, normal and high; the probability that the gauge gives the correct temperature is \mathbf{x} when it is working, but \mathbf{y} when it is faulty. Give the conditional probability table associated with \mathbf{G}

[continued on the next page]

c. Suppose the alarm works correctly unless it is faulty, in which case it never sounds. Give the conditional probability table associated with **A**

d. Suppose the alarm and gauge are working and the alarm sounds. Calculate an expression for the probability that the temperature of the core is too high, in terms of the various conditional probabilities in the network.

5. [Temporal Models] Knowledge Tracing (25 points)

A new online education website wants to provide individualized education. For simplicity imagine that users log into the website and continually answer randomly generated questions that cover triangle inequality. The only information accessible by the website is the score that the student gets on the problems they solve. Your task is to use these scores to track the extent to which a given student has mastered triangle inequality.

- (a) Is this problem better suited for a markov model or a hidden markov model?

- (b) Formalize this real world problem:
 - (i) Draw the temporal Bayesian network.

 - (ii) For each variable in your model give a brief description of its role and specify a **discrete** domain.

 - (iii) For each node in your model what are the dimensions of the CPT?

[continued on the next page]

- (c) We have invented a brain-measuring device that can measure exactly how well a student knows triangle inequality (You can call

```
getMastery()
```

at any point and the function will return a string in the domain {'low', 'medium', 'high'}).

Your task is to write python code that could use this device to learn the emission conditional probability table of scores given mastery.

$$p(score_i | mastery_i)$$

Your “learner” class will be given a constant stream of data from students solving problems. After all observations have completed, the `saveCpt` method will be called. Implement all functions.

```
class Learner():  
  
    # Function: Constructor  
    # -----  
    # Only if you need it  
    def __init__(self):
```

[continued on the next page]

```
# Function: Update
# -----
# Called once every time the student completes
# a problem. The score variable is the percent
# correct (0.0 to 1.0) of the student's
# solution. The studentId is passed in so that
# you can differentiate students.
def update(self, studentId, score):
```

[continued on the next page]

```
# Function: Save Cpt
# -----
# To save any data structure you can use
# util.saveProb( yourVariable, fileName).
# Save Cpt is Called once after several days of
# observing students solve triangle inequality
# questions. The variable you save should
# be a valid CPT (not a partial representation), though
# the format [dictionary, arrays] does not matter.
def saveCpt(self, fileName):
```