# CSE 415 Final Exam

Yuchen Wang

TOTAL POINTS

**145 / 200**

QUESTION 1

## A* Search 20 pts

### 1.1 (a) DFS order of nodes removed from OPEN 0 / 2

- **0 pts** Correct
- **1 pts** Partially Correct

✓ **- 2 pts Incorrect**

### 1.2 (b) BFS order of nodes removed from OPEN 0 / 2

- **0 pts** Correct
- **1 pts** Partially Correct

✓ **- 2 pts Incorrect**

### 1.3 (c) Formula, with quantifier, for admissibilty 2 / 4

- **0 pts** Correct
- **2 pts** Not using quantifier

✓ **- 4 pts Incorrect**

**+ 2 Point adjustment**

### 1.4 (d) Is hB admissible? Justify. 4 / 4

✓ **- 0 pts Correct**

- **2 pts** Partially correct; I'm very strict here. If you do not say if largest disk is not on peg R, it takes at least 1 or 2 moves, you get minus 2; if you say when the largest disk is on peg R, it takes at least 1 move, you get -1 because it could take 0 as in goal state.
- **4 pts** Incorrect

### 1.5 (e) A* order of nodes removed from OPEN 2 / 4

- **0 pts** Correct

✓ **- 2 pts Partially Correct**

- **4 pts** Incorrect

### 1.6 (f) Nodes expansions of BFS avoided by A* 0 / 4

- **0 pts** Correct
- **2 pts** Partially Correct

✓ **- 4 pts Incorrect**

QUESTION 2

## MDPs: Value Iteration 20 pts

### 2.1 (a) V1(s) for all s. 4 / 4

✓ **- 0 pts Correct**

- **1 pts** Mostly correct [there is a small error]
- **2 pts** Partially correct
- **3 pts** Mostly incorrect

+ **1 pts** shown a correct approach (partially) to answer

- **4 pts** No answer
- **4 pts** Incorrect answer

### 2.2 (b) V2(s) for all s. 5 / 8

- **0 pts** Correct
- **5 pts** Partial credit - more than half of the states are wrong
- **8 pts** No answer
- **7 pts** Partial credit - very few states are correct

✓ **+ 1 pts Shown a (partially) correct approach toward the answer**

- **8 pts** Incorrect answer
- **1 pts** Mostly correct (there is a small error)
- **6 pts** Partial credit - few states are correct

✓ **- 4 pts About half of the states are incorrect**

- **3 pts** Several states are incorrect

### 2.3 (c) V3(s) for selected states. 8 / 8

- **0 pts** Correct
- **8 pts** no answer

✓ + **1 pts** **shown some work towards the answer**

✓ + **2 pts** **D (correct approach)**

✓ + **2 pts** **E (correct approach)**

✓ + **1 pts** **A (correct approach)**

✓ + **1 pts** **F (correct approach)**

✓ + **1 pts** **I (correct answer)**

- **0 pts** correct
- **8 pts** incorrect answer

✓ - **8 pts** **reset rubic**

QUESTION 3

## Q-Learning 20 pts

### 3.1 Episode 1. 2 / 6

- **0 pts** Correct
- **6 pts** Incorrect or blank

✓ - **3 pts** **i1 is OK, but the rest are missing or incorrect.**

- **1 pts** Almost all correct.

✓ - **1 pts** **No justification for answer.**

- **1 pts** First 5 transitions are there, but too large.

### 3.2 Episode 2. 2 / 6

- **0 pts** Correct

✓ - **2 pts** **Although your i1 in episode 2 takes advantage of the 50 computed before, your e2 is not doing that.**

- **1 pts** First transition (from F towards D) should give -1. Prev. Q value was 0 in this direction from F.
- **2 pts** e2 should be 24.
- **6 pts** Blank or incorrect.

✓ - **2 pts** **First two updates are missing or incorrect.**

- **2 pts** i1 should be 75

### 3.3 Episode 3. 2 / 8

- **0 pts** Correct
- **1 pts** 1st update should give -1.5 for Q(F, phi5).
- **2 pts** 2nd update should give d3 = 10.5
- **2 pts** 3rd update should give e3 = 48.5

✓ - **8 pts** **blank or incorrect.**

- **5 pts** Some setup looks OK but values are wrong.
- **4 pts** Error(s) from episode 2 leading to bad values in episode 3.
- **1 pts** Missing last update, with value for Q(I, Exit) of 87.5

+ **2** **Point adjustment**

effort

QUESTION 4

## Joint Prob. Distr. and Bayes Nets 20 pts

### 4.1 (a) Numbers of free params in the Bayes Net 4 / 4

✓ - **0 pts** **Correct**

- **4 pts** should be 2, 3, 3. there are 8 in total
- **2 pts** there are two free parameters in A
- **2 pts** there are only 3 free parameters in B and C
- **2 pts** there are 2+3+3=8 in total

### 4.2 (b) Num. free params in a corresp. full joint. 2 / 2

✓ - **0 pts** **Correct**

- **2 pts** should be 3 * 2 * 2 - 1 = 11

### 4.3 (c) P(B=b1) 2 / 2

✓ - **0 pts** **Correct**

- **2 pts** should be 0.2 · 0.15 + 0.3 · 0.23 + 0.4 · 0.62

### 4.4 (d) P(C=c1) 2 / 2

✓ - **0 pts** **Correct**

- **2 pts** should be 0.1 · 0.15 + 0.45 · 0.23 + 0.95 · 0.62

### 4.5 (e) P(A=a1|B=B1) 4 / 4

✓ - **0 pts** **Correct**

- **4 pts** should be (0.2)(0.15)/P(B = b1) = (0.2)(0.15)/(0.2 · 0.15 + 0.3 · 0.23 + 0.4 · 0.62) by bayes rule
- **2 pts** denominator is not correct
- **2 pts** numerator is not correct, should be 0.2 * 0.15

### 4.6 (f) P(C=c1|B=b1). **0 / 6**

- **- 0 pts** Correct
- ✓ **- 6 pts** **should be P(C=c_1 | B = b_1) = sum_a P(C = c_1 | A = a) P(A = a | B = b_1)**
- **- 0 pts** B and C are not independent. They are only conditionally independent given A.

QUESTION 5

## Markov Models 20 pts

### 5.1 (a) P(R | F) **4 / 4**

- ✓ **- 0 pts Correct**
- **- 4 pts** Incorrect

### 5.2 (b) P(R_{t+2} | F_{t}) **6 / 6**

- ✓ **- 0 pts Correct**
- **- 2 pts** Formula OK except 2 wrong table entries
- **- 2 pts** Symbolic formula OK but numbers are off
- **- 1 pts** Minor arithmetic error
- **- 3 pts** Nontrivial error in formula
- **- 1 pts** Formula OK except 1 wrong table entry
- **- 6 pts** Incorrect or blank

### 5.3 (c) Stationary probability of R. **10 / 10**

- ✓ **- 0 pts Correct**
- **- 1 pts** Right approach, but using a wrong table entry.
- **- 3 pts** Two equations in setup are fine, but not properly using P(R)+P(F)=1.
- **- 1 pts** Mostly correct, with a small error in calculation or simplification.
- **- 10 pts** Blank or significantly incorrect
- **- 7 pts** Form has a slight resemblance to one of the needed equations.
- **- 5 pts** Partially correct setup
- **- 3 pts** Seems like a failure to use P(R)+P(F)=1.

QUESTION 6

## Hidden Markov Models 20 pts

### 6.1 (a) P0(s2) for t=0 **2 / 2**

- ✓ **- 0 pts Correct**
- **- 2 pts** Incorrect: It should be: 1−P0(s1) = 1−0.5 = 0.5
- **- 1 pts** Missing numbers! You at-least need to write the values of all the individual terms

### 6.2 (b) P1(s1) for t=1 **5 / 5**

- ✓ **- 0 pts Correct**
- **- 1 pts** Did not normalize the probabilities: 0.3/(0.3 + 0.1) = 0.75
- **- 3 pts** Incorrect! Value should be(before normalization) P0(s1)P(e3|s1) = 0.5□0.6 = 0.3. Then 0.75 after normalization
- **- 2 pts** Took the wrong emission. Should be e3
- **- 1 pts** Missing numbers! You at-least need to write the values of all the individual terms
- **- 5 pts** Nothing written

### 6.3 (c) P1(s2) for t=1 **5 / 5**

- ✓ **- 0 pts Correct**
- **- 1 pts** Did not normalize the probabilities: 0.1/(0.3 + 0.1) = 0.25
- **- 3 pts** Incorrect! Value should be(before normalization) P0(s2)P(e3|s2) = 0.5□0.2 = 0.1. Then 0.25 after normalization
- **- 2 pts** Took the wrong emission. Should be e3
- **- 1 pts** Missing numbers! You at-least need to write the values of all the individual terms
- **- 5 pts** Nothing written

### 6.4 (d) P2(s1) for t=2 **4 / 4**

- ✓ **- 0 pts Correct**
- **- 0 pts** Work looks correct! Given the incorrect values from previous parts
- **- 1 pts** Did not normalize
- **- 3 pts** Incorrect! Value should be(before normalization) P(e1|s1)□(P1(s2)P(s1|s2) +P1(s1)P(s1|s1)) = 0.25□(0.25□0.1 + 0.75□0.3) = 0.0625. Then 0.45 after normalization
- **- 2 pts** Took the wrong emission. Should be e1
- **- 1 pts** Missing numbers! You at-least need to write the values of all the individual terms
- **- 4 pts** Nothing written

### 6.5 (d) P2(s2) for t=2 4 / 4

✓ **- 0 pts** Correct

**- 0 pts** Work looks correct! Given the incorrect values from previous parts

**- 1 pts** Did not normalize

**- 3 pts** Incorrect! Value should be(before normalization) P(e1|s2)(P1(s2)P(s2|s2) +P1(s2)P(s1|s1)) = 0.1(0.250.9 + 0.750.7) = 0.075. Then 0.55 after normalization

**- 2 pts** Took the wrong emission. Should be e1

**- 1 pts** Missing numbers! You at-least need to write the values of all the individual terms

**- 4 pts** Nothing written

QUESTION 7

## Perceptrons 20 pts

### 7.1 (a) Weights and threshold for NAND gate 7 / 7

✓ **- 0 pts** Correct

**- 2 pts** Only used one negated weight

**- 4 pts** Missing both negated weights

**- 3 pts** Incorrect threshold

**- 7 pts** No answer

### 7.2 (b) Training (3 cycles) 9 / 9

✓ **- 0 pts** Correct

**- 3 pts** Step 1 incorrect

**- 3 pts** Step 2 incorrect

**- 3 pts** Step 3 incorrect

**- 9 pts** No answer

**- 2 pts** Treated ground truth as part of the data vector

### 7.3 (c) Find a possible training vector for given outcome 4 / 4

✓ **- 0 pts** Correct

**- 2 pts** Incorrect learning rate

**- 4 pts** Incorrect

**- 4 pts** No answer

**- 1 pts** Correct vector off by constant factor

QUESTION 8

## Natural Language Processing 20 pts

### 8.1 (a) Eleven scores 4 / 4

✓ **- 0 pts** Correct

**- 1 pts** Almost correct 9 or 10 out of 11

**- 2 pts** Some right answers: 6, 7, or 8 correct.

**- 3 pts** Mostly wrong (3, 4, or 5 correct)

**- 4 pts** Wrong, or fewer than 3 correct.

**- 2 pts** Easy logs not taken. Examples were shown in solutions to the A6 problem.

### 8.2 (b) Parse tree 1 3 / 3

✓ **- 0 pts** Correct

**- 2 pts** Illegal use of a production.

**- 3 pts** Incorrect or blank

### 8.3 (c) Parse tree 2 3 / 3

✓ **- 0 pts** Correct

**- 3 pts** Incorrect or blank

### 8.4 (d) Total score and probability for parse 1 4 / 4

✓ **- 0 pts** Correct

**- 1 pts** Score off by 1 or 2

**- 2 pts** Score off by more than 2 or missing.

**- 2 pts** Probability missing, inconsistent with score or incorrectly determined from score.

### 8.5 (e) Total score and probability for parse 2 4 / 4

✓ **- 0 pts** Correct

**- 1 pts** Score is 1 or 2 off

**- 2 pts** Score is more than 2 off, missing, or incorrect in another way.

**- 2 pts** Probability is inconsistent with score, missing, or incorrectly determined from score.

### 8.6 (f) More probable parse 2 / 2

✓ **- 0 pts** Correct

**- 2 pts** Incorrect or clearly a guess without all the supporting data.

QUESTION 9

## Multiple choice 40 pts

### 9.1 (a) Graph search: main advantage of BFS vs DFS **5 / 5**

✓ **- 0 pts Correct**

**- 5 pts** Incorrect

### 9.2 (b) Expectimax most appropriate if ... **0 / 5**

**- 0 pts** Correct

✓ **- 5 pts Incorrect**

### 9.3 (c) A* fails when **0 / 5**

**- 0 pts** Correct

✓ **- 5 pts Incorrect**

### 9.4 (d) IDDFS -- what's NOT an advantage **0 / 5**

**- 0 pts** Correct

✓ **- 5 pts Incorrect**

### 9.5 (e) Perceptron training: worst obstacle **5 / 5**

✓ **- 0 pts Correct**

**- 5 pts** Incorrect

### 9.6 (f) One of Asimov's laws of robotics **5 / 5**

✓ **- 0 pts Correct**

**- 5 pts** Incorrect

### 9.7 (g) Not part of Kurzweil's "singularity" **0 / 5**

**- 0 pts** Correct

✓ **- 5 pts Incorrect**

### 9.8 (h) Obstacles to using Asimov's laws for self-driving cars **5 / 5**

✓ **- 0 pts Correct**

**- 5 pts** Incorrect

# CSE 415–Winter 2019 — Final Examination

by the Staff of CSE 415, Winter 2019

**INSTRUCTIONS:** Read these instructions carefully. Write your full name at the top of this cover page. Use the official version of your name that appears in the course registration. Make sure you have all 12 pages. **Also write your name on top of every page or it is possible that you would not receive credit for that page.** (We will be removing staples from the exams, and if a page of yours gets lost due to not having your name on it, you could miss credit for that page.) Write your UW student number and your UWNetID where shown below.

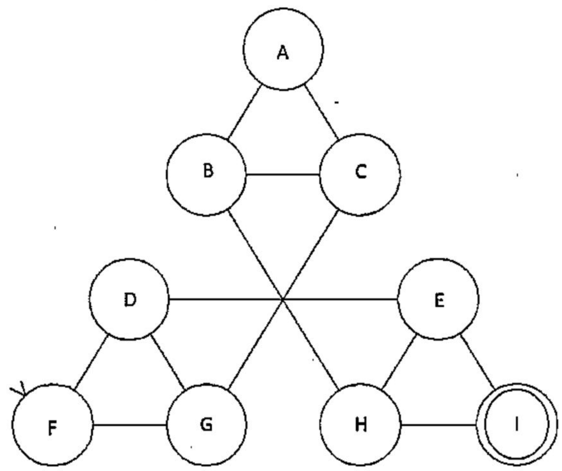
Write legibly and if you use a pencil, make sure that you write darkly enough that a normal scanner will pick up your writing. If you need more space, use the margins. Do all problems. This is a CLOSED-BOOK, CLOSED-NOTES examination. Do not use any books, notes, calculators, or other electronic devices. There are 8 questions worth 20 points each plus one question that has 8 multiple choice parts, and worth 40 points. The total of possible points is 200. Each problem has multiple parts, and the allocations of points among the parts are as shown on each individual problem. Note that some problems may be significantly easier or more time-consuming than others. However, the overall time to complete the exam is estimated to be 90-100 minutes for a student familiar with the material.

**POLICY ON QUESTIONS AND CLARIFICATIONS:** Do not ask questions during the exam. Instead, if you find one of the exam problems to be ambiguous or unclear, state your objection here on the cover page, clearly identifying the problem number and part number, as well as describing your issue (and continue in the margin on the page of the problem, if necessary). If your objection is accepted by the grader, then points may be adjusted in your score.

| | |
|---|---|
| UW Student Number: | 1826728 |
| UWNetID | markp6 |

# 1 A* Search (Jifan)

The graph below shows the state space for a 2-disk version of the Towers of Hanoi puzzle. The start state is F (both disks on the left) and the goal state is I. (The top node of the graph is for state A, where both disks are in the middle.) Assume the three pegs of the puzzle are L (left), M (middle), and R (right).




Assume the set of operators is the following, and that they are tried in the given order when generating successors:

1. $\phi_0$: Try to move a disk from R to M.
2. $\phi_1$: Try to move a disk from R to L.
3. $\phi_2$: Try to move a disk from M to R.
4. $\phi_3$: Try to move a disk from M to L.
5. $\phi_4$: Try to move a disk from L to R.
6. $\phi_5$: Try to move a disk from L to M.

It might be helpful to think of the operators as moving in particular directions on the graph, according to the following scheme:

| $\phi_0$ | $\phi_1$ | $\phi_2$ | $\phi_3$ | $\phi_4$ | $\phi_5$ |
|---|---|---|---|---|---|
| ↖ | ← | ↘ | ↙ | → | ↗ |

(a) (2 points) What is the order in which nodes are REMOVED from the OPEN list in Depth-First Search?

F D E I

(Just as in the loop-based formulation of DFS presented in lecture, we assume the list L of successors, after deleting nodes already on CLOSED, is pre-pended onto the OPEN list, and any member of L already on OPEN has been removed from OPEN.)

Assume each search (DFS, BFS, A*) terminates as soon as the goal node I is removed from OPEN and found to be a goal node.

(b) (2 points) What is the order in which nodes are REMOVED from the OPEN list in Breadth-First Search?

FDGEI

(c) (4 points) In this and the following parts, we'll assume that the cost of each edge in the problem-space graph above is 1. Assume $h(s)$ represents the smallest number of moves needed to get from state $s$ to the goal state (all disks on peg R). Using a formula, give a necessary and sufficient condition for a heuristic function $h_i$ to be admissible. Use a quantifier (either $\forall$ or $\exists$, as appropriate) in your formula. The set of all states for the puzzle is represented as $\Sigma$.

$h_i(\forall s \in \Sigma\, s) \leq$ actual distance from s to goal state.

(d) (4 points) Consider the heuristic function defined as follows:

$$h_B(s) = 0 \text{ if the largest disk is on peg R; } 1, \text{ otherwise.}$$

Is $h_B$ admissible? Justify your answer.

Yes, it is, because actual distance is $\geq 0$ if largest disk on R, otherwise actual distance is $\geq 1$.

(e) (4 points) What is the order in which states are removed from the OPEN list by the A* algorithm using heuristic $h_B$? (Break ties in priority values by the same ordering from BFS: the order in which states are placed onto the OPEN list.)

FDEI

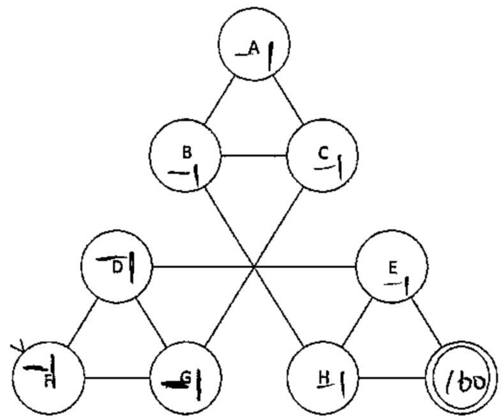
(f) (4 points) What node expansions, if any, are saved here by A*, versus BFS? (A node expansion is the generation of all that state's successors, whether or not any of those successors already exist.)

## 2 MDPs: Value Iteration (Hamid)

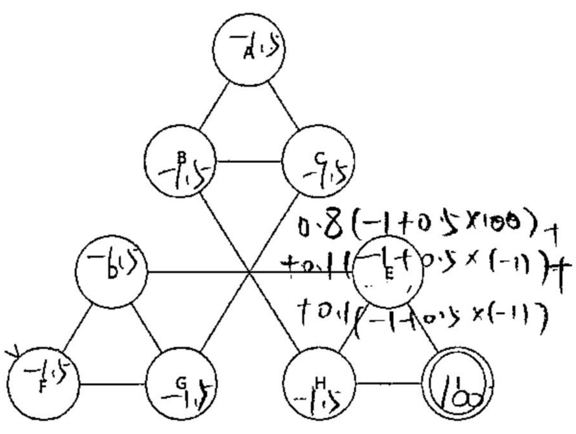
Once again, we are considering the Towers-of-Hanoi puzzle having 2 disks. Our job is now to consider a more general context in which the intended moves sometimes lead to possibly undesired states. Assume we have an MDP whose states are those shown (A through I) plus a terminal state Z (not shown in the diagrams), reached from exit state I. Allowed actions are only those corresponding to graph edges, except that from state I, only the transition to Z is allowed.

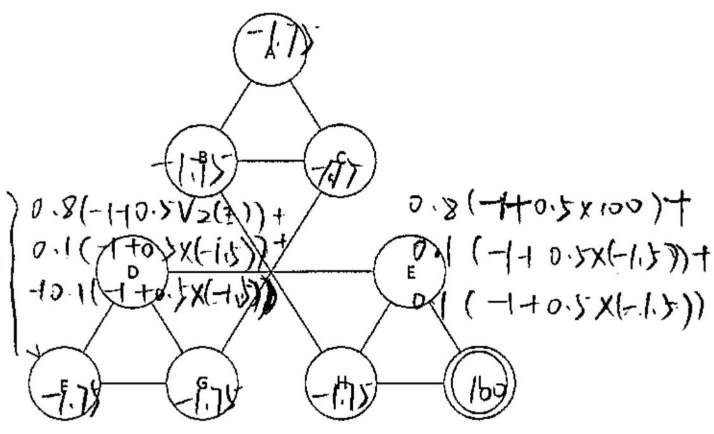
All transitions are deterministic except for those starting in D and E, where there is an 0.8 of going in the intended direction and 0.1 probability of going in each of the other two directions. Assume the living cost is $R = -1$ in any transition except I to Z, where the reward is 100. Assume the agent's discount factor is $\gamma = 0.5$. We assume $V_0(s) = 0$ for all $s$.

(a) (4 points) Compute $V_1(s)$, and write the values in their states below.




(b) (8 points) Compute $V_2(s)$, and write the values in their states below on the left copy of the graph. For $V_2(E)$, an expression involving numbers, multiplications and additions will be sufficient.




(c) (8 points) Compute $V_3(s)$ for only the following states, and write the values in their states above on the right copy of the graph: A, F, D, E, I. You can just give correct expressions for $V_3(D)$ and $V_3(E)$ ; we don't require that you calculate the final number.

# 3 MDPs: Q-Learning (Jifan)

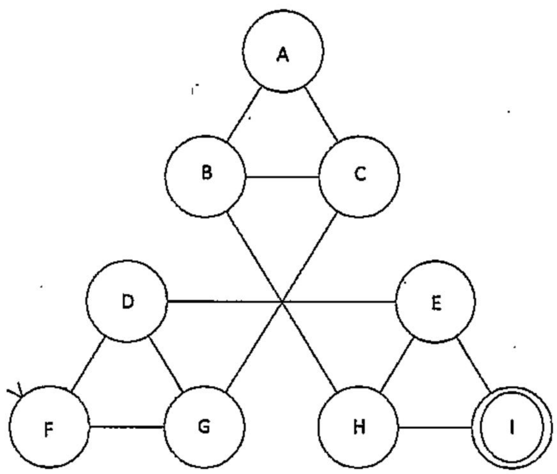
Let's now consider what an agent does in the same Towers-of-Hanoi world (expressed as a Markov Decision Process again), when it is not given the MDP parameters T and R, but must learn about the MDP through its own experience. The agent's discount factor here is: $\gamma = 1$. (No discounting)

Use the following Q-update rule after each transition of the following episodes to adjust the Q values (all initially 0), and show updated values next to the outgoing edges from each node. Use a learning rate $\alpha = 0.5$. For example, the first transition in episode 1 should result in a value of $-1$ coming out of state F towards G, since the sample has value $-2$ (because the reward is $-2$, the maximum Q value out of G is 0), the existing $Q(F, \phi_4)$ is 0, and the learning rate is 0.5.

$$Q(s,a) \leftarrow \alpha \textbf{ sample } + (1-\alpha)Q(s,a)$$

$$\text{where } \textbf{sample } = r + \gamma \max_{a'} Q(s', a')$$

Here the transition being processed is [ $s, a, s', r$ ].




Here are the episodes. Show all updates, but extra space is provided for details on some.

1. (6 points) [F, $\phi_4$, G, $-2$], [G, $\phi_5$, C, $-2$], [C, $\phi_1$, B, $-2$], [B, $\phi_2$, H, $-2$], [H, $\phi_4$, I, $-2$], [I, Exit, Z, 100]

   Show your computation for $i_1 = Q(I, Exit)$ after the transition from I.

   $i_1 =$ 50

2. (6 points) [F, $\phi_5$, D, $-2$], [D, $\phi_4$, E, $-2$], [E, $\phi_2$, I, $-2$], [I, Exit, Z, 100]

   Show your computation for $e_2 = Q(E, \phi_2)$ after the transition from E.

   $e_2 =$ $-1$

   Show your computation for $i_2 = Q(I, Exit)$ after this episode's transition from I.

   $i_1 =$ $0.5\, Q(I, Exit) + 0.5(100 + 1 \times 0) = 75$

3. (8 points) [F, $\phi_5$, D, $-2$], [D, $\phi_4$, E, $-2$], [E, $\phi_2$, I, $-2$], [I, Exit, Z, 100]

   Show your computation for $d_3 = Q(D, \phi_4)$ after this episode's transition from D.

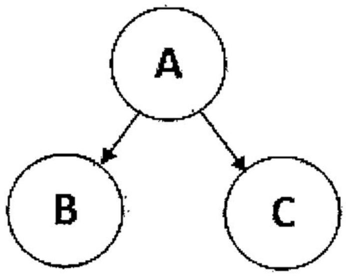
   $d_3 =$ $0.5\, Q(D, \phi_4) + 0.5(-2 + 1 \times (-1)) = -1.5$

   Show your computation for $e_3 = Q(E, \phi_2)$ after this episode's transition from E.

   $e_3 =$ $0.5 \times (-1.5) + 0.5(-2 + 1 \times (-1)) = -2.25$

# 4 Joint Probability Distributions and Bayes Nets (Sam)

In this example, we consider a joint probability distribution of three random variables with different domains. We use a Bayes net to represent this distribution. The graph and conditional probability tables for the Bayes net are shown below.




| $A$ | $P(A)$ |
|---|---|
| $a_1$ | 0.15 |
| $a_2$ | 0.23 |
| $a_3$ | 0.62 |

| $B$ | $A$ | $P(B\|A)$ |
|---|---|---|
| $b_1$ | $a_1$ | 0.2 |
| $b_2$ | $a_1$ | 0.8 |
| $b_1$ | $a_2$ | 0.3 |
| $b_2$ | $a_2$ | 0.7 |
| $b_1$ | $a_3$ | 0.4 |
| $b_2$ | $a_3$ | 0.6 |

| $C$ | $A$ | $P(C\|A)$ |
|---|---|---|
| $c_1$ | $a_1$ | 0.1 |
| $c_2$ | $a_1$ | 0.9 |
| $c_1$ | $a_2$ | 0.45 |
| $c_2$ | $a_2$ | 0.55 |
| $c_1$ | $a_3$ | 0.95 |
| $c_2$ | $a_3$ | 0.05 |

In the following, wherever "compute" is requested, give a formula that shows what numbers and operations are required, but don't actually do the arithmetic. For example:

$$P(A = a_1, B = b_1 | C = c_1) = (0.15)(0.2)(0.1)/(0.1 + 0.5 + 0.9)$$

(a) (4 points) What are the numbers of free parameters in each of the three tables? 2 3 3
What is the total number of free parameters among the tables? 8

(b) (2 points) If the joint distribution for $A$, $B$, and $C$ were general and could not be factored as this Bayes net, how many free parameters would it have? 11

(c) (2 points) "Compute" $P(B = b_1)$. $0.2 \times 0.15 + 0.3 \times 0.23 + 0.4 \times 0.62$

(d) (2 points) "Compute" $P(C = c_1)$. $0.1 \times 0.15 + 0.45 \times 0.23 + 0.95 \times 0.62$

(e) (4 points) "Compute" $P(A = a_1 | B = b_1)$. $\dfrac{0.2 \times 0.15}{0.2 \times 0.15 + 0.3 \times 0.23 + 0.4 \times 0.62}$

(f) (6 points) "Compute" $P(C = c_1 | B = b_1)$. For this example, you may wish to define one or more other probabilities (and "compute" them) to simplify your main expression.

$$P(C=c_1 | B=b_1) = \frac{P(B=b_1 | C=c_1) P(C=c_1)}{P(B=b_1)} = \frac{P(B=b_1, C=c_1)}{P(B=b_1)}$$

$$P(B=b_1, C=c_1) = 0.15 \times 0.2 \times 0.1 + 0.23 \times 0.3 \times 0.45 + 0.62 \times 0.4 \times 0.95$$

$$P(C=c_1 | B=b_1) = \frac{0.15 \times 0.2 \times 0.1 + 0.23 \times 0.3 \times 0.45 + 0.62 \times 0.4 \times 0.95}{0.2 \times 0.15 + 0.3 \times 0.23 + 0.4 \times 0.62}$$

# 5 Markov Models (Sam)

Assume we model the stock market with a two-state Markov model. On each day, the market is either rising ($R$) or falling ($F$). The transition matrix is this:

| $X_t$ | $X_{t+1}$ | $P(X_{t+1}\|X_t)$ |
|---|---|---|
| $R$ | $R$ | 0.8 |
| $R$ | $F$ | 0.2 |
| $F$ | $R$ | 0.3 |
| $F$ | $F$ | 0.7 |

(a) (4 points) Assuming the market is falling today, determine the probability that it will be rising tomorrow.

$P(X_{t+1} = R \mid X_t = F) = 0.3$

(b) (6 points) Assuming the market is falling today, determine the probability that it will be rising the day after tomorrow. (It may or may not be rising tomorrow.)

$P = 0.3 \times 0.8 + 0.7 \times 0.3$

$= 0.45$

(c) (10 points) Find the stationary probability $r = P_\infty(R)$ that the market is rising.

$r = P(R|R)r + P(R|F)(1-r)$

$r = 0.8 \times r + 0.3(1-r)$

$r = 0.6$

# 6 Hidden Markov Models (Divye)

In this exercise, we will run the Forward algorithm for 2 steps, for the following given Hidden Markov Model.

Transition model:

| $X_t$ | $X_{t+1}$ | $P(X_{t+1}\|X_t)$ |
|---|---|---|
| $s_1$ | $s_1$ | 0.3 |
| $s_1$ | $s_2$ | 0.7 |
| $s_2$ | $s_1$ | 0.1 |
| $s_2$ | $s_2$ | 0.9 |

Emission model:

| $X_t$ | $e_1$ | $e_2$ | $e_3$ |
|---|---|---|---|
| $s_1$ | 0.25 | 0.15 | 0.6 |
| $s_2$ | 0.1 | 0.7 | 0.2 |

Let's say the initial probability for $s_1$ is $P_0(s_1) = 0.5$ (at $t = 0$), and we observed $e_3$ at $t = 1$ and $e_1$ at $t = 2$.

In each of the following, an unevaluated expression involving operations and numbers will be sufficient; you don't need a calculator.

(a) (2 points) What is $P_0(s_2)$ for $t = 0$?

$P_0(s_2) = 0.5$

(b) (5 points) What is $P_1(s_1)$ (for $t = 1$)? (This is after the observation of $e_3$.)

$P_1(s_1) = P(e_3|s_1)\,P_0(s_1) = 0.3$

normalize: $P_1(s_1) = 0.75$

(c) (5 points) Similarly, what is $P_1(s_2)$ (for $t = 1$)?

$P_1(s_2) = P(e_3|s_2)P_0(s_2) = 0.1$

normalize $P_1(s_2) = 0.25$

(d) (4 points) What is $P_2(s_1)$ (for $t = 2$)?

$P_2(s_1) = P(e_1|s_1)\big(P(s_1|s_2)P_1(s_2) + P(s_1|s_1)P_1(s_1)\big)$

$= 0.25 \times (0.1 \times 0.25 + 0.3 \times 0.75)$

normalize: $P_2(s_1) = \frac{5}{11}$

(e) (4 points) What is $P_2(s_2)$ (for $t = 2$)?

$P_2(s_2) = P(e_1|s_2)\big(P(s_2|s_1)P_1(s_1) + P(s_2|s_2)P_1(s_2)\big)$

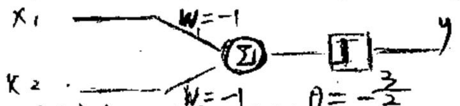
$= 0.1\,(0.7 \times 0.75 + 0.9 \times 0.25)$

normalize $P_2(s_2) = \frac{6}{11}$

# 7 Perceptrons (Rob)

For the following questions, assume all perceptrons output 1 iff their weighted input $a$ satisfies $a > \theta$, and 0 otherwise. (This is not $a \geq \theta$.)

(a) (7 points) Write the weights and threshold for a two-input perceptron that would cause it to function as a NAND gate (that is, the negation of the logical AND). Possible input values are $\{0, 1\}$, where 0 represents False, and 1 represents True.




(b) (9 points) Assuming an initial weight vector of $\langle 1, 1 \rangle$ and static $\theta = 0$, train this perceptron with the following example data. Use a learning rate of 1. Write the weight vector that would result after each training example. There is no bias term for this perceptron. Expected output for each example is listed after the data vector:

(i) $(\langle 0, 0 \rangle, 0)$ $\langle 1, 1 \rangle$

(ii) $(\langle -2, 3 \rangle, 1)$ $\langle 1, 1 \rangle$

(iii) $(\langle 0, 1 \rangle, 0)$ $\langle 1, 0 \rangle$

(c) (4 points) Assume we initially had a perceptron with weight vector $\langle 0, 0, 0 \rangle$ with $\theta = 1$. Then we gave a positive training example that was misclassified as negative and resulted in our weight vector changing to $\langle 1, 0, 2 \rangle$. Give a possible training data vector and learning rate that could cause this result.

training data:
$(\langle 1, 0, 2 \rangle, 1)$

learning rate is 1

# 8 NLP (Steve)

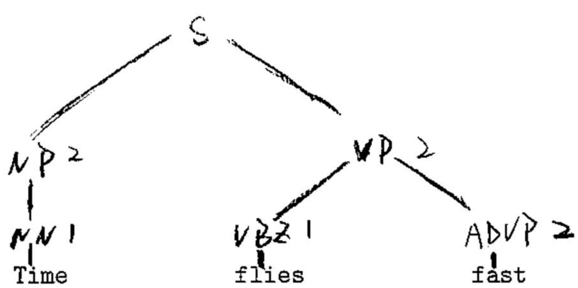
Consider the sentence "Time flies fast." This might mean that the clock seems to move quickly, or it might mean that some kind of insects are not eating.

With the given probabilistic context-free grammar, find two legal parses, and compute a score for each one. Then identify the most probable parse using the scores. The grammar is given below. Consider the number at the right of a production to be the conditional probability of applying that production given that the symbol to be expanded is the symbol on the left-hand side of the production. Convert each probability into a score by taking score $= -\log_2(p)$.

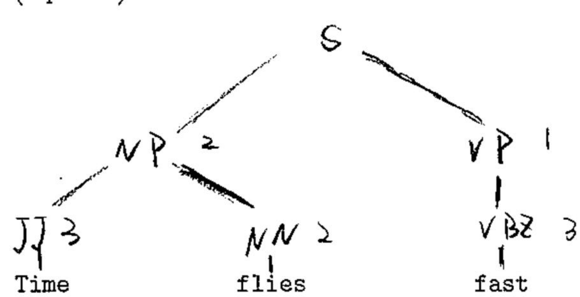
| | | | | |
|---|---|---|---|---|
| S | ::= | NP VP | 1.00 | 0 |
| NP | ::= | JJ NN | 0.25 | 2 |
| NP | ::= | NN | 0.25 | 2 |
| VP | ::= | VBZ | 0.50 | 1 |
| VP | ::= | VBZ ADVP | 0.25 | 2 |
| NN | ::= | Time | 0.50 | 1 |
| JJ | ::= | Time | 0.125 | 3 |
| NN | ::= | flies | 0.25 | 2 |
| VBZ | ::= | flies | 0.50 | 1 |
| VBZ | ::= | fast | 0.125 | 3 |
| ADVP | ::= | fast | 0.25 | 2 |

(a) (4 point) Scores for each production rule: write the scores next to the probabilities above.

(b) (3 points) Parse number 1:




(c) (3 points) Parse number 2:




(d) (4 points) Total score 8 and overall probability $2^{-8}$ for parse number 1.

(e) (4 points) Total score 11 and overall probability $2^{-11}$ for parse number 2.

(f) (2 point) Which parse is more probable? 1

# 9 Multiple Choice (Rob)

This question is worth 40 points (5 for each subquestion). For each subquestion, underline the letter of the *best* answer. For example: A.

(a) In graph search with a constant branching factor $b \geq 2$, what is the main advantage of BFS over DFS?

- (A) Guarantee of using less memory to find a shortest path to the goal than with Depth-First Search?
- (B) Guarantee of identifying a shortest path upon arrival at a goal state.
- (C) By processing the OPEN list like a queue rather than a stack, the generation of successors naturally goes faster.
- (D) By searching in all directions equally, there is no chance that the search algorithm will fail to reach a goal if one is reachable.

(b) Expectimax search is particularly appropriate in adversarial search if

- (A) the opponent is rational but always trying to minimize the state score.
- (B) the opponent is irrational and unpredictable.
- (C) the opponent is rational and always trying to maximize the state score.
- (D) the opponent always makes a predictable move.

(c) A* search will fail to find a goal node

- (A) if its heuristic is not admissible.
- (B) only if no goal node is reachable.
- (C) if its heuristic is not consistent.
- (D) only if no goal node exists.

(d) In tree search, which is not an advantage of iterative deepening search over straight depth-first search to the same maximum depth?

- (A) In game-playing, it can provide an anytime algorithm for computing a good move within the available time.
- (B) When searching for a goal, when it does find a goal, it can immediately identify a shortest path.
- (C) Its memory requirements are no greater, and on average, less.
- (D) It might take longer to find a goal node.

(e) Perceptron training can be viewed as a kind of search for a certain type of hyperplane. What is most likely to prevent finding a solution?

(A) A learning rate that is constant and equal to 1.

(B) A training set in which the positive and negative examples cannot be completely separated by any hyperplane.

(C) An iterative algorithm that has no limit on the number of iterations it might use during training.

(D) The existence of extra dimensions in the vector representation of the data items, such that these dimensions contain irrelevant features.

(f) Which of these is one of Asimov's Three Laws of Robotics?

(A) People must respect the autonomy of any real robot.

(B) A robot must protect its own existence unless that conflicts with the other laws.

(C) People and robots must stay within their own separate realms of expertise.

(D) No robot may consume a resource such as food, which is a resource needed by humans for survival.

(g) Which of these is not part of Kurzweil's "singularity"?

(A) Robots are expected to fully control humans.

(B) Robots will have been improving their own designs.

(C) Great strides will have been made in neuroscience.

(D) Artificial intelligence will help spread human values.

(h) Which of these stand in the way of using Asimov's rules for self-driving cars?

(A) A car cannot be a robot, and vice versa.

(B) Robots cannot always predict the consequences of their actions accurately.

(C) The rules are written in English and therefore irrelevant to robots and computers.

(D) The rules need to be adopted by state legislatures before they can be applied.