Q1 Score: **11.0** Total Score: **76.0**

1- True/False (15 points)

For each of the statements below, fill in the bubble T if the statement is always and unconditionally true, or fill in the bubble F if it is always false, sometimes false, or just does not make sense: (1pt each)

1	●	F
2	T	●
3	●	F
4	●	F
5	●	F
6	T	●
7	●	F
8	●	F
9	T	●
10	●	F
11	T	●
12	●	F
13	●	F
14	T	●
15	T	●

1- In a well formed Bayesian Belief Network, a node is always conditionally independent of its non-descendants given its parents.

2- Overfitting can result when a machine learning hypothesis space has too few dimensions.

3- An advantage of using decision trees for machine learning is that the classifiers produced can be easily implemented with rules.

4- A and B are independent if and only if P(A|B) = P(A)

5- We can use dynamic programming to solve MDPs.

6- People usually use forward error propagation to train neural networks.

7- Bayes' rule contains both prior and posterior probabilities.

8- If P(A|B,C)= P(B|A,C) then P(A|C)=P(B|C)

9- Value Iteration Algorithm can calculate optimal values iteratively using Dynamic Programming in MDP problems.

10- Reinforcement learning is used when the model of world is unknown and/or rewards are delayed

11- Q-learning eliminates the need for an explicit transition model

12- Given the hypothesis ***Bird*** = **WarmB & LayE & Fly** for the concept **Bird**, then **f(Robin: WarmB & LayE & Fly)=Bird** is a True Positive

13- Given the hypothesis ***Bird*** = **WarmB & LayE** for the concept **Bird**, then **f(Ostrich: WarmB & LayE & ¬Fly)=Bird** is a False Negative

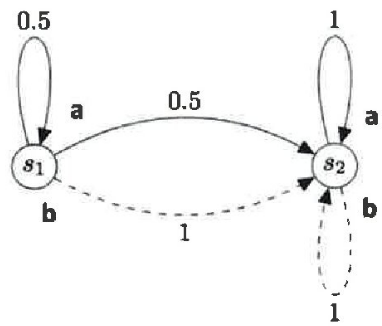
14- Given the hypothesis ***Bird*** = **WarmB & LayE** for the concept **Bird**, then **f(Platypus: WarmB & LayE & ¬Fly)=Mammal** is a False Positive

15- Given the hypothesis ***Bird*** = **WarmB & LayE** for the concept **Bird**, then **f(Bat: WarmB & ¬LayE & Fly)=Mammal** is a True Positive

Q2 Score: **5.0**

2- Markov Decision Process (20 points)

Consider a simple MDP with two states, s1 and s2, and two actions, a (solid line) and b (dashed line); the numbers indicate transition probabilities. Rewards, which just depend on state and action (not the state resulting from the action), are shown in the table below.



$R(S_1\|a) = 8$	$R(S_2\|a) = -4$
$R(S_1\|b) = 16$	$R(S_2\|b) = -4$

Supposing that U_0 of both states is 0 and the discount factor, γ, is .5, fill in the four boxes (U_1 and U_2), Show all your work (including formulas) below.

Remember $U_{t+1}(s) = R(s) + \text{Max}_{a\varepsilon A} \{\gamma \Sigma_{s'\varepsilon S} P(s'|a,s)\, U_t(s')\}$

$U_0(s_1)=0$	$U_0(s_2)=0$
$U_1(s_1)=8$	$U_1(s_2)=-4$
$U_2(s_1)=10$	$U_2(s_2)=-2$

S_1 can only be got by S_1. through action a.

$U_1(S_1) = R(S_1|a) + 0.5 \times 0.5 \times U_0(S_1) = 8$

$U_2(S_1) = R(S_1|a) + 0.5 \times 0.5 \times U_1(S_1) = 10$

S_2 can be accessed through action a or b.

$U_1(S_2)$: since $U_0(S_1) = U_0(S_2) = 0$, $U_1(S_2) = R(S_2) = -4$

$U_2(S_2)$: for action a, $\gamma \Sigma_{s'\varepsilon S} P(s'|a,s)\, U_1(s') = 0.5(0.5 \times 8 + 1 \times -4) = 0$

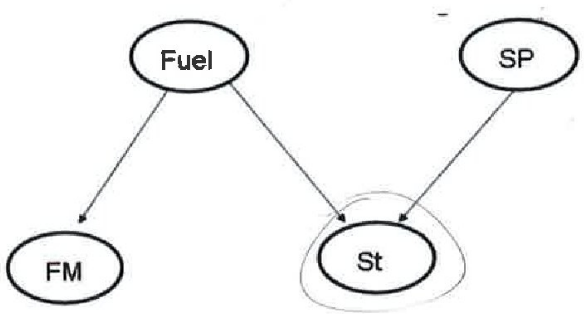
for action b, $\gamma \Sigma_{s'\varepsilon S} P(s'|b,s)\, U_1(s') = 0.5(1 \times 8 + 1 \times -4) = 2$.

$\therefore U_2(S_2) = R(S_2) + 2 = -2$

Q3 Score: **6.0**

3- Bayesian Networks (20 points)

The following Bayesian network captures some of the causal dependencies pertaining to whether your professor's car will start in the morning:



The random variables *Start?* (St), *Fuel?* (Fuel), and *Clean-Spark-Plugs*? (SP) can each take on the values Yes or No, while *Fuel-Meter* (FM) can take on the values Full, Half, or Empty. You know P(Fuel=Yes) = 0.6 and P(SP=YES) = 0.8 . You also know the following conditional probability tables (CPTs) for P(FM | Fuel) and P(St | SP) :

Fuel	P(Fm=Full)	P(Fm=Half)	Empty
Yes	0.4	0.4	0.2
No	0.05	0.1	0.85

Fuel	SP	P(St=Yes)
Yes	Yes	0.95
Yes	No	0.1
No	Yes	0.01
No	No	0

1- Use Bayes rule to compute P(Fuel = Yes | FM = Empty).

$P(Fuel = Yes \mid FM = Empty)$

$= P(FM = Empty \mid Fuel = Yes) \times P(Fuel = Yes) \ / \ P(FM = Empty)$

$= (1-0.4-0.4) \times 0.6 \ / \ [0.6 \times (1-0.4-0.4) + (1-0.6) \times (1-0.05-0.1)]$

$\doteq 0.283$

Q4 Score: **14.0**

0.48
- 0.0348

0.99×0.48

2- Write down an expression for the full joint distribution over all the four random variables as a product of conditional probabilities given the above Bayesian network structure.

$P(FM, Fuel, St, SP)$
$= P(FM \mid Fuel) \cdot P(St \mid Fuel, SP) \cdot P(Fuel) \cdot P(SP)$

3- Compute the joint probability P(Fuel = No, SP = Yes, FM = Half, St = No).

$P(Fuel = No, SP = Yes, FM = half, St = No)$
$= P(half \mid \neg Fuel) \cdot P(\neg St \mid \neg Fuel, SP) \cdot P(\neg Fuel) \cdot P(SP)$
$= 0.1 \times (1 - 0.01) \times (1 - 0.6) \times 0.8$
$= 0.04752$

4- Use the full joint distribution in (b) and *inference by enumeration* to compute the probability that the car will start given that the fuel meter says "Empty." Show all the steps involved.

$P(St \mid FM = Empty)$
$= \alpha \cdot P(St, FM = Empty)$
$= \alpha \cdot \sum_{Fuel} \sum_{SP} P(St, FM = Empty, Fuel, SP)$
$= \alpha \cdot \sum_{Fuel} \sum_{SP} P(Empty \mid Fuel) \cdot P(St \mid Fuel, SP) \cdot P(Fuel) \cdot P(SP)$
$= \alpha \cdot (0.2 \times 0.95 \times 0.6 \times 0.8 + 0.2 \times 0.1 \times 0.6 \times 0.2 + 0.85 \times 0.01 \times 0.4 \times 0.8)$
$= \dfrac{0.09352}{0.6 \times 0.2 + 0.4 \times 0.85}$
$\doteq 0.16$

0.01
0.12
0.58
0.58
0.12
116
58
0.0792
0.00272
0.09352

2.95
$\times$ 0.6
0.57
0.85
0.32
170
255
0.00272

56
57

Q5 Score: **20.0**

4- Decision Trees (20 points)

Suppose a problem domain is described by the attributes A, B, and C, where A and B can each assume the values *Yes* or *No*, and C can assume the values *Yes*, *No*, or *Maybe*. Based on the decision tree learning algorithm discussed in class and in the textbook (best attribute at each step chosen according to information gain), construct a decision tree for this problem using the following set of training examples:

Example	A	B	C	Output
1	No	Yes	Yes	Yes
2	No	No	Maybe	No
3	Yes	No	No	No
4	Yes	Yes	Maybe	Yes
5	Yes	No	Yes	Yes
6	No	No	Yes	No

$$I(\frac{p}{p+n},\frac{n}{p+n}) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n}$$

$$remainder(A) = \sum_{i=1}^{v}\frac{p_i+n_i}{p+n} I(\frac{p_i}{p_i+n_i},\frac{n_i}{p_i+n_i})$$

$$IG(A) = I(\frac{p}{p+n},\frac{n}{p+n}) - remainder(A)$$

step1: choose first root:

$IG(A) = I(\frac{3}{3+3},\frac{3}{3+3}) - (\frac{3}{3+3} I(\frac{1}{3},\frac{2}{3}) + \frac{3}{3+3} I(\frac{2}{3},\frac{1}{3}))$

$= 1 + (\frac{1}{3}\log_2\frac{1}{3} + \frac{2}{3}\log_2\frac{2}{3})$

$IG(B) = I(\frac{2}{6},\frac{4}{6}) - (\frac{2}{6} I(\frac{2}{2},\frac{0}{2}) + \frac{4}{6} I(\frac{1}{4},\frac{3}{4}))$

$= -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3} + \frac{2}{3}(\frac{1}{4}\log_2\frac{1}{4} + \frac{3}{4}\log_2\frac{3}{4})$

$IG(C) = I(\frac{3}{6},\frac{2}{6},\frac{1}{6}) - (\frac{3}{6} I(\frac{2}{3},\frac{1}{3}) + \frac{2}{6} I(\frac{1}{2},\frac{1}{2}) + \frac{1}{6} I(1,0))$

$= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{3}\log_2\frac{1}{3} - \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{2}(\frac{2}{3}\log_2\frac{2}{3} + \frac{1}{3}\log_2\frac{1}{3})$

$+ \frac{1}{3}$

$\because IG(B) > IG(C) > IG(A)$

$\therefore$ first root is B.

step 2: choose senvond root:

$IG'(A) = I(\frac{2}{4},\frac{2}{4}) - (\frac{2}{4} I(\frac{2}{2},0) + \frac{2}{4} I(\frac{1}{2},\frac{1}{2})) = \frac{1}{2}$

$IG'(C) = I(\frac{2}{4},\frac{1}{4},\frac{1}{4}) - \frac{2}{4} I(\frac{1}{2},\frac{1}{2})$

$= -\frac{2}{4}\log_2\frac{1}{2} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{2}$

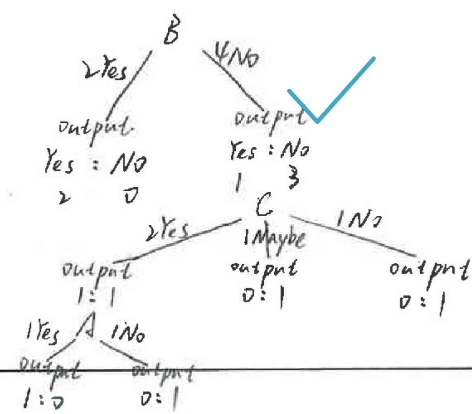
$= \frac{1}{2} + \frac{1}{2} + \frac{1}{2} - \frac{1}{2}$

$= 1$

$\because IG'(C) > IG'(A)$

$\therefore$ second root is C

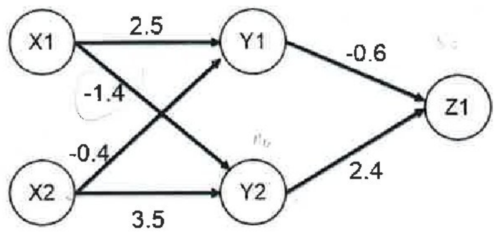
final decision tree:



Q6 Score: **10.0**

5- Neural Networks (15 points)

In this network the input layer consists of units X1 and X2. The hidden layer is Y1 and Y2, and the output layer is Z1. Units Y1, Y2 and Z1 all have threshold functions where t > 0 for the unit to activate.



1- Complete the outputs for the given inputs in the table below:

X1	X2	Output
1	1	1
1	0	0
0	1	1
0	0	0

2- Can a weight be changed for the network to compute the Boolean function OR? If yes, which weight and what value should it be changed to? If no, why?

Yes. change -0.6 or -1.4 into any number > 0.

3- Can a single weight be changed for the network to compute the Boolean function XOR? If yes, which weight and what value should it be changed to? If no, why?

No. Because XOR is not linear separable

Q7 Score: **6.0**

6- AI Applications (10pts)

1- Choose 2 AI applications presented for Participation Exercise 3 and write a 1 sentence description of each application:

Application 1: Cecus: it is the first solved complete AI to play pocker well

Application 2: Robot controlled remotely through brain thoughts: one of the most popular front-edge Brain-Machine Interface to help disabled people to contact with the world.

2- How does each application represent knowledge about the world?

Cecus: cards are represented by numbers, knowledge is represented using rules in first-order logic.

Remote Robot: location and environment detected by GPS and a camera, knowledge represented in first-order logic.

3- How does each application integrate new percepts to past experience?

Cecus: It has inner state to record previous rounds. New percepts or actions will be based on knowledge and previous experience. It learns through CRR+ algorithm, which help him fill out strategys to win.

Remote Robot: It has inner state to record previous orders from the pilot and previous actions has done. It learns through RNN (Recurrent Neural Network) to deal with signals from thoughts, promoting the accuracy of decoding.

Q8 Score: **4.0**

4- What are the space and time complexity for each application?

Cecus: the number of pocker cases is more than 10^8, and he needs to remember the expierience. so the space complexity is expotional. About the time complexity, since he has efficient algorithm, it is near linear.

Remote Robot: space complexity is expotional, since he needs not only the location itself, but also location of other objects, and plevions action need to be recorded. Time-complexity is linear.

5- Which application will scale better and why?

Cecus will scale better, because the environment he is faced with is simpler than the other, so is the logic. It is currently a solved problem and cecus can perform well even given a time limit per round. The CRR-plus algorithm promotes its efficiency very well. On the other hand, human thoughts are more complex for Remote Robot to deal with, and he will never know what will happen around himself. He need to deal with situation he has never seen (e.g. strange obstacles)