NAME________________________
login:________________________
Signature: ________________________

Computer Science and Engineering 151
Introduction to Artificial Intelligence
June 10, 2004
FINAL EXAM
DO NOT TURN THIS PAGE UNTIL YOU ARE TOLD TO START!!!!

- Please DO NOT put your name at the top of each page:
  - This should prevent residual sexism, racism, favoritism lurking in the unconscious of your professor & TA from biasing grading!!
- THE EXAM IS CLOSED BOOK. IN FACT, PLEASE LEAVE YOUR BOOKS AT THE FRONT OF THE CLASS!
- Once the exam has started, SORRY, NO TALKING!!!
- No, you can't even say "later, dude!"
- There are 10 problems: Make sure you have all of them - AFTER I TELL YOU TO START!
- Read each question carefully.
- Remain calm at all times!

Problem	Type	Points	Score
1	True/False	10	
2	Fill in the blanks, short answer	10	
3	Bayes' Rule	10	
4	Uncertainty	10	
5	Bayes Networks	10	
6	Probabilistic reasoning over time	10	
7	Markov Decision Processes	10	
8	Reinforcement Learning I	10	
9	Reinforcement Learning II	10	
10	Neural nets: representation	10	
11	Neural nets: learning	10	
	Total	100	

True/False

1. (10 pts: +1 for correct, -1 for incorrect, 0 for no answer)
   If you would like to justify an answer, feel free.

______ The advantage of the value iteration algorithm is that it always returns the true values of the states even with partial information about the world.

______ The transition model describes the probability of moving to a state s if action a is taken.

______ The difference between a passive agent and an active agent is that the passive agent does not actually interact with the world.

______ The basic idea of TD methods are to make neighboring states fit the constraints of the Bellman equations on average.

______ In the real world problem of helicopter flying, reinforcement learning methods perform almost as well as expert humans.

______ The perceptron learning algorithm can be used to learn any linearly separable problem.

______ Direct sampling methods such as MCMC are used to learn the parameters for the networks, but not the structure of the networks.

______ Bayesian networks encode conditional independence relationships between random variables.

______ Assuming that a person's preferences follow the six axioms of utility theory, then there is a function U that maps states to real numbers such that $U(s) > U(s')$ if the person prefers s to s'.

______ The Perceptron Convergence Procedure described in class only changes weights when the network makes a mistake.

______ Any function a multiple layer perceptron (back propagation network) can represent, it can learn.

______ The NETTalk network described in class was trained using initial weights of 0.

______ It is unclear where the hidden units are hidden, and who put them there. However, learning in a backprop net consists of finding them.

2. Short Answer

2a. Write the product rule (i.e. $p(a, b) = ?$). Show how you can derive Bayes rule from this.

2b. What is the learning rate parameter? How does it affect the performance of an iterative learning method? Besides a constant value, what is another reasonable way to set the learning rate?

2c. Define the Q function used in Q learning.

2d. Describe one of the "tricks" described in class to speedup convergence for backpropagation.

3. Multiple Choice

3a. The joint probability table of two binary events, A and B, requires ______ to specify.

1. 1 number
2. 2 numbers
3. 3 numbers
4. 4 numbers

3b. One key idea of Bayesian networks is

1. to make conditional independence assumptions, which can easily be represented graphically;
2. to represent bayes' rule in each node
3. to represent the full joint probability table in graphical form
4. to represent probabilities as a graph.

3c. Back propagation works by following the gradient of

1. the sum squared error with respect to the inputs
2. the sum squared error with respect to the weights
3. the outputs with respect to the inputs
4. the weights with respect to the sum squared error

3d. The back propagation algorithm learns the weights to hidden units as well as outputs. Thus, the hidden units

1. learn to detect useful features of the input for the task.
2. never amount to much.
3. are themselves computing nonlinearly separable functions.
4. become gaussians.

3e. The form of one rule for learning (I'm not saying which one) is:

1. $net_i \ = \ \sum_j w_{ij} * z_j$
2. $\Delta w_{ij}[t+1] \ = \ \eta \delta_i z_j \ + \ \mu \Delta w_{ij}[t]$
3. $\Delta w_{ij} \ = \ \eta * activation_i * net_j$
4. $\Delta output_i \ = \ (teacher - output_i) * output_j$
5. $\Delta w_{ij} \ = \ (output_i) * (1 - output_i) \sum_k w_{ki} * \delta_k * output_j$

Markov Decision Processes

7a. (3pts) Describe precisely the three components that define a Markov decision process.

7b. (1 pt) What is the Markovian assumption?

7c. (1 pt) Describe the difference between fully observable MDPs and partially observable MDPs. Be specific.

7d. (4pts) The two algorithms that were described in the book for MDP's are value iteration and policy iteration. Briefly describe how each of these work. Are these algorithms guaranteed to find the optimal policy? If so, under what conditions?

7e. (1 pt) Write down the formula for the optimal policy, given the correct values V for each state and the correct transition function T.

Reinforcement Learning I: Exploration functions

8a. (1 pt) What is an exploration function within the framework of reinforcement learning?

8b. (2 pts) What is the purpose of an exploration function?

8c. (7 pts) Describe two different exploration functions forms, including one that you make up. Briefly describe the advantages and disadvantages of using each of these functions for Q-learning.

Reinforcement Learning II

You're the TA for an AI course. You need to come up with an assignment for the class. You decide to make the implement TD-Checkers. To help the students along, you get them started with some information about the problem. Describe the the states and actions for this problem.

Do you need an exploration function? Explain your answer.

How many inputs are there to the network? (Note: a checkerboard is 8 by 8). What is a good guess for the appropriate number of hidden units to use for a neural network? Make sure to justify your choice.

Describe two possible (i.e. implementable) activation functions for the network (including small graphs). How many output units are required for each of these activation functions for this problem?

Your students will need the deriviatives of these two activation functions. You decide to be a nice TA and calculate these. Show them here.

Learning: Perceptrons (5 points)
6. The following is a table describing the learning in a perceptron as we did in class. As in class, we use a *bias* instead of a threshold. So the activation rule is:

```
if net_input + bias >= 0  then 1
                          else 0
```

This means we can use the standard learning rule and the bias works as usual, as a weight from an input whose activation is always 1.0. Fill in the next 2 lines of the table.
You have to remember the learning rule, assume a learning rate of 1.0, as we did class. The output produced on any row is a function of the weights on that row, but the weights on the *next* row correspond to the *new weights* generated by the error on the previous row.

Input1	Input2	Input3	Weight1	Weight2	Weight3	Bias	Output	Teacher
0	1	0	0	0	0	0	1	1
1	0	0	0	0	0	0	1	0
1	1	0	-1	0	0	-1	0	1
0	0	0						0
1	1	0						1

Learning: Back propagation (15 points)

7(a) (12 pts). Compute the δ's for the output unit and the hidden units after the following pattern presentation. The network structure is at the left (recall δ for an output is (t-o) times the slope, and assume standard sigmoids). The second-from-left column shows the network weights, with the biases shown at the nodes. The middle column shows the net input to a node. The next column shows the activations of the inputs, hiddens and outputs. Your job: Fill in the last column. PLEASE SHOW YOUR WORK. YOU WILL GET PARTIAL CREDIT FOR CORRECT FORMULAS. YOU ARE ALLOWED TO ROUND TO TWO SIGNIFICANT DIGITS FOR EACH COMPUTATION. (Note: These numbers are *made up*: the actual activations may not match the net_inputs).

```
                                                     teacher = 1.0
              |  weights/biases    |   net_input    |  activation  |      delta       |
              |                    |                |              |                  |
              |                    |                |              |                  |
   OUT        |       -.30         |     -0.4       |      .4      |  OUT:___         |
   / \        |       /   \        |                |              |                  |
  /   \       |      1     -.25    |                |              |                  |
 /     \      |     /         \    |                |              |                  |
H1      H2    |  0.00          .25 |  -2       2    |  .10     .9  |H1:___   H2:___   |
| \    / |    |   |  \        / |  |                |              |                  |
|  \  /  |    |  -1    -1   1  .75 |                |              |                  |
|   X    |    |   |      X      |  |                |              |                  |
|  / \   |    |   |     / \     |  |                |              |                  |
IN1    IN2    |                    |                |  1.0    1.0  |                  |
```

7(b) (3pts). Finally, given the δ's you computed, what will the new weight from hidden unit 1 (H1) to the Output (OUT) (currently 1) be, if no momentum is being used and the learning rate is 1.0?