CS 540-1: Introduction to Artificial Intelligence

Final Exam

2:45-4:45pm, December 20, 1995

Room 3425 Sterling Hall

CLOSED BOOK

(one page of notes allowed)

Write your answers on these pages and show your work. If you feel that a question is not fully specified, state any assumptions that you need to make in order to solve the problem. You may use the backs of these sheets for scratch work.

Write your name on this and all other pages of this exam. Make sure your exam contains eight problems on ten pages.

Name ______________________________

Student ID ______________________________

Problem	**Score**	**Max Score**
1	_____	20
2	_____	25
3	_____	20
4	_____	20
5	_____	20
6	_____	10
7	_____	20
8	_____	15
Total	_____	150

PROBLEM 1 - Decision Trees (20 points)

(a) Assume you are given the following three features of TV shows, with the possible values shown, and wish to learn how to predict future top-10 shows.

Type	$\in$	{Comedy, Drama, News, Sports}
Location	$\in$	{LA, NYC, Various}
Duration	$\in$	{30min, 60min}

Assume that the ID3 algorithm is given the following set of categorized examples from previous television seasons. Calculate the decision tree that ID3 would produce. *Show all your work.* (You may use the abbreviations that are used to describe the examples.) Break ties by preferring the feature with the *least* number of possible values.

T = C	L = NYC	D = 30	+
T = C	L = LA	D = 60	-
T = N	L = Var	D = 60	+
T = N	L = LA	D = 30	+
T = S	L = Var	D = 60	-
T = D	L = LA	D = 60	-

lg is the base 2 log

lg(a*b) = lg(a) + lg(b)
lg(a/b) = lg(a) - lg(b)

lg(1) = 0
lg(2) = 1
lg(3) = 1.58
lg(4) = 2
lg(5) = 2.32
lg(6) = 2.58

(b) Briefly explain the space ID3 searches and the search strategy it uses.

What is the initial state?

What is the goal test?

PROBLEM 2 - Neural Networks (25 points)

(a) Hand ‘‘wire’’ (i.e., manually configure) a perceptron so that it computes the Boolean AND of its two inputs.

(b) Now consider using the delta rule to *train* a perceptron to learn the AND function. Assume the initial weights and threshold of your perceptron are all –0.1 and that the learning rate (i.e., α or η) is 0.2. Be sure to *train the threshold.*

Draw the initial state of this perceptron.

Draw the state of the perceptron after training it on the input *In1=1 & In2=1*.

(c) Assuming you trained it on all possible inputs enough times, could the delta rule learn the AND function? Explain your answer using *feature space*.

(d) Assuming *In2* = 0 and the threshold = –0.1, draw the *weight space* (and associated error surface) for the AND function (by using specific values for *In2* and the threshold, you do not have to draw these two dimensions, since they are constant).

(e) Discuss the claim that the backpropagation algorithm uses a ‘‘hill climbing’’ search strategy.

(f) Dr. I. M. Wright is given 10,000 hospital records describing the initial symptoms of patients who, after weeks of treatment with a new drug, were or were not cured. Due to the availability of alternate treatments, it would be useful to be able to accurately predict, based on the initial symptoms of an incoming patient, whether or not they should receive the new drug. Dr. Wright decides to use a neural network to learn this function.

Dr. Wright uses the following methodology. First, he randomly chooses 1000 patient records and marks them as ‘‘testset’’ cases. Next, he uses the remaining 9000 examples to train five different neural networks; these networks have 0, 5, 10, 15, 20 hidden units, respectively. Once training completes, he tests the five networks on the held-aside testset and finds that the one with 15 hidden units has the best error rate (it’s 97.2% accurate). Based on these results, Dr. Wright sends to the hospital the network with 15 hidden units and tells the doctors there that if they use this neural network they can expect an diagnosis error rate of 2.8%.

Explain the major methodological flaw that Dr. Wright made.

PROBLEM 3 - Prolog (20 points)

(a) Consider the Prolog program below.

```
c(1,4).
c(2,1).
c(2,3).
c(2,6).
c(3,5).
c(5,4).
r(X,X).
r(X,Y) :- c(Z,Y), r(X,Z).
```

What would Prolog return if given the following queries? (Report *all* the answers returned, in the order Prolog would return them.)

```
?- r(2,4).
```

```
?- r(W,4).
```

(b) Write a Prolog program for *sick(Person,Year)* that captures the following world knowledge:

> *John and Mary get sick every year.*
> *Pebbles, the child of Fred and Wilma, will get sick in 1996.*
> *Every year Maggie sits next to Pebbles on the school bus.*
> *People who sit next to sick people will get sick* [that same year].
> *Parents catch* [that same year] *every cold their children get.*
> *Having a cold is one way of getting sick.*

The predicate *sick(Person,Year)* means that *Person* will be (was) sick during *Year*. You may use, without needing to further define them, the predicates: *human(Person), parentOf(Parent,Child), hadFluShot(Person,Year), hasCold(Person,Year), hasFlu(Person,Year), sitsNextTo(PersonA,PersonB,Year)*.

PROBLEM 4 - Search (20 points)

For each of the following assertions, draw a *specific* search graph that supports it. (If you wish to use an infinite search graph, it is fine to use ‘‘ . . . ’’ to indicate the remainder of an infinite subgraph.) Explain your answers. Note: assume that best-first search solely uses $f=h$ as its scoring function.

(a) *Best-first search can explore more nodes than breadth-first search.*

(b) *Hill climbing can produce a solution when depth-first search does not.*

(c) *A* can produce a lower-cost solution path than does best-first search when both algorithms use the same admissible h function.*

PROBLEM 5 - Predicate Calculus (20 points)

Represent the following English sentences using predicate calculus. In the first two sentences, be sure to represent *move* and *cook* as situation-calculus operators. If you feel a sentence is ambiguous, you need only give one FOPC representation, but also provide an unambiguous paraphrase. Be sure to choose reasonable constants, predicates, and functions.

Unless the box currently on the conveyor belt is marked fragile,
Robbie the robot will move it from the warehouse to the shipping dock.

Cooking a pizza does not change its ingredients.

John is the neighbor of Mary's sister.

To be a respected butterfly collector, one must possess two distinct and rare butterflies.

Everyone on the team owes everyone else on the team an apology.

PROBLEM 6 - FOPC Interpretations (10 points)

Create a simple *formal interpretation* that shows the following wff is *not* valid.
Explain your solution.

$$\{\forall\ x\ [P(x) \Rightarrow Q(x)]\ \} \Rightarrow \{\forall\ y\ [P(y) \wedge Q(y)]\ \}$$

PROBLEM 7 - Miscellaneous Questions (20 points)

Answer the following *true* (T) or *false* (F). Provide brief justifications of your answers.

Depth-first search will always terminate in finite search spaces.

Ans: ___ Just:

The crossover operator plays a central role in genetic algorithms.

Ans: ___ Just:

Every wff that can be derived via a ‘‘natural deduction’’ proof can also be proved using resolution.

Ans: ___ Just:

Negation-by-failure is a logically sound inference rule.

Ans: ___ Just:

Resolution theorem proving will not terminate if the theorem one is trying to prove is false.

Ans: ___ Just:

Fuzzy logic reduces to standard logic in the case where all propositions are either completely true (ie, have value 1) or completely false (have value 0).

Ans: ___ Just:

The wff's P(?x,?x) *and* P(f(?a,?b),f(?b,?a)) *do not unify.*

Ans: ___ Just:

PROBLEM 8 - Lisp (15 points)

Assume you have started a ‘‘fresh’’ Lisp. What would the following s-expressions return? Write ‘‘error’’ if Lisp would complain when evaluating an s-expr. In cases where multiple s-expressions are in a group, you need only report the returned value of the *last* s-expr.

```
(cons 1 (list 2 3))
```

```
(append (1 2 3) (4 5 6))
```

```
(rest (first (rest ’( (1) (2) (3) ))))
```

```
(assoc 1 ’( (1 2) (1 1) (2 1) ))
```

```
(if (member 3 ’( 1 (2) ((3)) )) ’yes ’no)
```

```
(setf x nil)
(dolist (y ’(1 2 3)) (setf x (cons y x)))
x
```

```
(defun f(x y)
   "a mystery function"
   (if (< x 1)
       (+ 1 (f (- x 1) y))
      y))
(f 5 1)
```