CS 540-1: Introduction to Artificial Intelligence

Final Exam: 2:45-4:45pm, December 16, 1994
Room B239, Van Vleck

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 seven problems on ten pages.

Name _____

Student ID _____

Problem	**Score**	**Max Score**
1	_____	10
2	_____	35
3	_____	15
4	_____	20
5	_____	15
6	_____	40
7	_____	15
Total	_____	150

PROBLEM 1 - Planning (10 points)

Consider the following STRIPS operators:

Operator 1		*Operator 3*	
preconditions:	$A \wedge C$	*preconditions:*	B
add list:	$B \wedge D$	*add list:*	D
delete list:	C	*delete list:*	C
Operator 2		*Operator 4*	
preconditions:	$A \wedge D$	*preconditions:*	$A \wedge C$
add list:	E	*add list:*	B
delete list:	A	*delete list:*	E

Assume the task is to construct a plan that correctly converts the initial state $A \wedge B \wedge C$ into the final state $D \wedge E$

Show the initial state and its *immediate* descendants (i.e., neighbors) in the search space when:

---

you are using standard state-based search to solve the problem

---

you are using the STRIPS algorithm (goal-directed planning)

---

you are using the TWEAK algorithm (non-linear planning)
[you do not have to answer this precisely; simply illustrate the gist of TWEAK]

PROBLEM 2 - Neural Networks (35 points)

Consider a very simple perceptron, one with a *single* input unit. Assume the following are your training examples.

Input	*Output*
-5	1
4	1
7	1
-1	0
0	0
2	0

Part A. Draw below the *feature space* for this task. Can this data set be learned (separated) by a perceptron? Explain your answer.

Part B. Draw below the *weight space* (i.e., error on the data set as a function of the value of the network's weight) for this problem. For simplicity, assume *for Part B only* that the threshold is always zero (i.e., assume you are *not* training the threshold); since it is constant, you do not need to include the threshold in your weight-space graph.

Part C. Now consider using the delta rule to train a perceptron on this data; this time you should consider *training the threshold*. Assume that the weight and the threshold are both initially -0.1, and let d=0.1. Draw the perceptron at the times requested below.

---

before training

---

after training on the first example above (explain the changes, if any)

---

after training on the second example above (explain the changes, if any)

Part D. Which search-control strategy does backpropagation use (e.g., depth-first search, beam search, best-first search, etc.)? What about ID3? Explain your answers.

PROBLEM 3 - Decision Trees (15 points)

Consider learning a decision tree that you could use to judge whether or not you will like a given restaurant. Assume you have chosen to use the following three features to describe restaurants, with the possible values shown.

Price ∈ *{Low, Med, High}*
Type ∈ *{Hamburgers, Pizza, Fish, Vegetarian}*

Assume Quinlan's ID3 algorithm is given the following set of classified training examples. Calculate the decision tree that ID3 would produce. *Show all your work.* (You may use the abbreviations that are used to describe the examples.)

P = L	T = H	+
P = L	T = V	+
P = M	T = F	-
P = M	T = V	+
P = H	T = P	-

---

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

PROBLEM 4 - Knowledge Representation (20 points)

Part A. Represent the following English sentences using first-order predicate calculus. In the first sentence, be sure to represent *paint* as a situation-calculus operator. If you feel a sentence is ambiguous, you need only give one FOPC representation, but also provide an unambiguous paraphrase.

Painting a block does not change its weight.

At least one question on the CS540 final is hard.

Whenever it snows in Madison, there is a slippery intersection somewhere in town.

Part B. Draw a simple AND-OR tree that (partially) represents the knowledge involved in deciding to move north in the Agent World. Include at least two AND nodes and one OR node.

PROBLEM 5 - Resolution Theorem Proving (15 points)

Consider the following statements.

Every man has his eyes checked by the doctor.

No man checks his own eyes.

Everyone is either a man or a woman.

Part A. Represent the English sentences above in FOPC.

Part B. Represent the above FOPC in clausal form.

Part C. Using resolution, show that the doctor is a woman.

PROBLEM 6 - Miscellaneous Questions (40 points)

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

Breadth-first search is guaranteed to find the optimal solution.

Ans: _____ Just:

Hill climbing and beam search, when the beam width is set to one, will always produce the same answer.

Ans: _____ Just:

The horizon effect *does not impact* hill-climbing *algorithms.*

Ans: _____ Just:

The qualification problem *is an important issue in planning.*

Ans: _____ Just:

Using negation by failure *is one way to address the* frame problem.

Ans: _____ Just:

There exists an interpretation *that shows that* $\forall x$ [pilot(x) $\rightarrow$ drives_cars(x)]
does not represent ‘‘Some pilots drive cars.’’

Ans: _____ Just:

If wff’s A and B unify and wff’s B and C unify, then wff’s A and C unify.

Ans: _____ Just:

If working memory = { g(a,a), g(a,b), g(b,a), g(b,c), h(b,a), h(c,a), h(c,c), h(c,b), h(b,c) }
then the following precondition of a rule is satisfiable: g(X,Y) $\wedge$ g(Y,Z) $\wedge$ h(X,Z).

Ans: _____ Just:

The use of backward-chaining *reasoning eliminates the need for a* conflict-resolution strategy.

Ans: _____ Just:

If learning algorithm A’s accuracy on the training data is higher than that of learning algorithm B’s, then algorithm A is definitely better for this data set.

Ans: _____ Just:

PROBLEM 7 - Common Lisp (15 points)

Assume you have started a *fresh* Lisp. What would the following Lisp expressions return? (If you think a Lisp error would result, write ERROR.)

```
> (append '(1 2 3) (cons 1 nil))
```

```
> (* 4 (+ 5 3) (* (+ 2 4) 2))
```

```
> (list (append '(1 2)
                (rest (cons (first '(3)) '((4))))
                ))
```

```
> (defun mystery (x y)
     "A mystery function."
     (cond ((null x) y)
           (t (append (mystery (rest x) y)
                      (list (first x))))))
MYSTERY
> (mystery '(1 2) '(a b))
```

```
> (setf x '(1 2 3))
(1 2 3)
> (setf (second x) 'new)
NEW
> x
```

The end - have a good vacation!