CS 540: Introduction to Artificial Intelligence

Final Exam: 2:45-4:45pm, May 10, 1999
Room 168 Noland

CLOSED BOOK
(two sheets of notes and a calculator allowed)

Write your answers on these pages and show your work. If you feel that a question is not fully specified, state any assumptions 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 nine pages.

Name ____

Student ID ____

Problem	**Score**	**Max Score**
1	______	**15**
2	______	**13**
3	______	**17**
4	______	**16**
5	______	**10**
6	______	**12**
7	______	**10**
8	______	**7**
TOTAL	______	**100**

Problem 1 – First-Order Predicate Calculus (15 points)

Convert each of the following English sentences into First-Order Predicate Calculus, using reasonably named predicates, functions, and constants. If you feel a sentence is ambiguous, clarify which meaning you're representing in logic.

Someone is taller than Mary.

Red blocks weigh more than blue ones.

Except for BillG, for every person there exists someone richer.

If you give something of yours to someone else, they own it. [You must use *situation calculus* here.]

Borrowing a book doesn't change who owns it. [You must use *situation calculus* here.]

Problem 2 – Production Systems (13 points)

Consider the production-system rules listed below. Variables are indicated by a leading '?' and all other terms are constants. The conflict-resolution strategy is to choose the rule with the *most* preconditions, with the extra constraint that no rule can be used more than once *with the same variable bindings*. The predicate'£' is *procedurally defined*, and this production system uses *negation-by-failure*.

(1) IF P(?x, ?y) Ù Q(?x, a, ?z) Ù Ø Q(?z, a, ?z)
THEN assert R(?x)
retract Q(?x, a, ?z)

(2) IF R(?x) Ù Q(?x, ?y, ?z)
THEN retract R(?x)

(3) IF P(f(?x), ?y) Ù ?x £ ?y Ù Q(f(?w), ?w, f(?z)) Ù ?y £ ?z
THEN print SUCCESS!

(4) IF R(?x)
THEN assert Q(f(?x), ?x, f(?x))

Let the initial contents of working memory be:

P(f(1), 2)	*P(3, f(4))*	*P(4, 4)*
Q(f(1), a, f(1))	*Q(4, a, 3)*	
R(f(1))		

Fill in the table below, showing the first three (3) cycles of the production system's operation.

Cycle	*Matching Rules*	*Chosen Rule*
i		
ii		
iii		

List the *net changes* to working memory that result from the above three cycles:

Problem 3 – Bayesian and Case-Based Reasoning (17 points)

Imagine you wish to recognize bad “widgets” produced by your factory. You’re able to measure two numeric properties of each widget: *P1* and *P2*. The value of each property is discretized to be one of *{low (L), normal (N), high(H)}*. You randomly grab 5 widgets off of your assembly line and extensively test whether or not they are good, obtaining the following results:

P1	*P2*	*Result*
L	N	good
H	L	bad
N	H	good
L	H	bad
N	N	good

Explain how you could use this data and Bayes’ Rule to determine whether the following new widget is more likely to be a good or a bad one (be sure to show your work and explain any assumptions/simplifications you make):

L	L	?

How might a case-based reasoning system address this task? How would it handle the test example above?

Problem 4 – Neural Networks (16 points)

Using a *1-of-N* encoding, show how a perceptron could be configured to address the "widget" task of Problem 3. Initialize all the weights to 0 and the threshold to 0.1.

Show how one would apply the the delta rule to each of the first two training examples in Problem 3 (show the state of the perceptron *after each example*). Use the *step function* as the activation function and let *good=1* and *bad=0*. Set η (the learning rate) to 0.25. Be sure to adjust the threshold during training.

Assuming that all the other weights and the threshold have value 0 and using the training set from Problem 3, draw the *weight space* for the weight between the *P1=L* input node and the output node.

Problem 5 – Genetic Algorithms (10 points)

Consider the following fitness function:

$$\textit{fitness} = 5a + 3bc - d + 2e$$

where *a-e* are all Boolean-valued parameters. Compute the fitness of each of the members of the initial population below. Also compute the probability that each member of the population will be selected during the *fitness-proportional reproduction* process.

a	*b*	*c*	*d*	*e*	*fitness*	*reproduction probability*
1	1	0	1	1		
0	1	1	0	1		
1	1	0	0	0		
1	0	1	1	1		
1	0	0	0	0		

Assuming the first two of members of the population are selected for reproduction, and the *cross-over* point is that between the *b* and the *c*, show the resulting children:

Problem 6 – Miscellaneous Short Answers (12 points)

Briefly describe each of the following AI concepts:

Heuristic Functions

Hidden Markov Models (HMMs)

Linear Separability

*A**

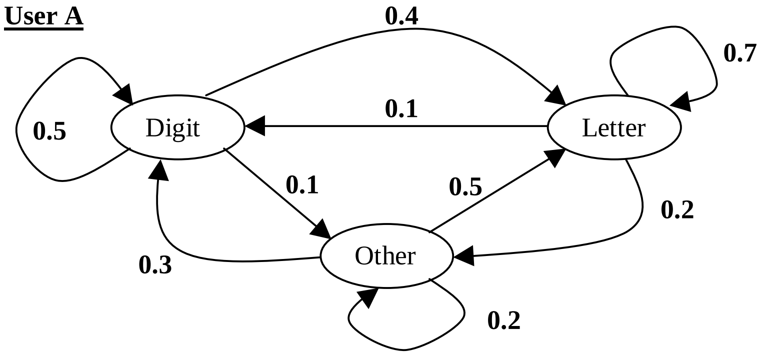
Problem 7 – Markov Models (10 points)

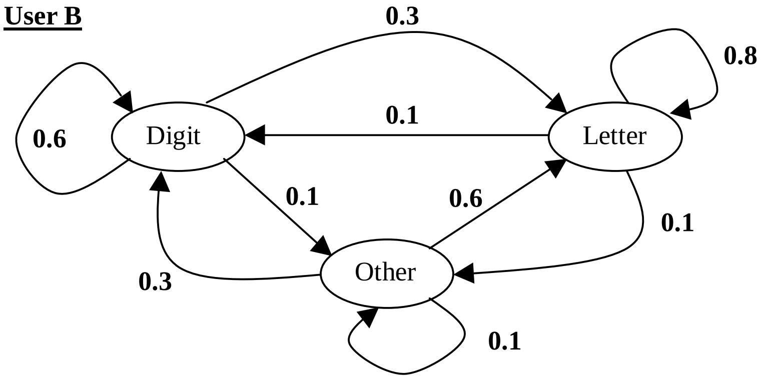
Assume that *User A* and *User B* equally share a computer, and that you wish to write a program that determines which person is currently using the computer. You choose to create a (first-order) Markov Model that characterizes each user's typing behavior. You decide to group their keystrokes into three classes and have estimated the transition probabilities, producing the two graphs below. Both users always start in the *Other* state upon logging in.

Now imagine that the current user logs on and immediately types the following:

> *IOU $15*

Who is more likely to be the current user? Show and explain your calculations.





Problem 8 – Gradient Descent on an Error Function (7 points)

Derive the weight-change rule for a perceptron where the following is the error function and the activation function of the output unit is the *identity* function (i.e., the output equals the incoming weighted sum and no threshold is involved).

$$Error \;^{o}\; output - 5$$