SPRING 1998
COMPUTER SCIENCES DEPARTMENT
UNIVERSITY OF WISCONSIN – MADISON
Ph.D. QUALIFYING EXAMINATION

Artificial Intelligence

MONDAY, FEBRUARY 2
3:00 – 7:00 PM
ROOM 1263 COMPUTER SCIENCE AND STATISTICS

GENERAL INSTRUCTIONS:

1. Answer each question in a separate book.
2. Indicate on the cover of *each* book the area of the exam, your code number, and the question answered in that book. On *one* of your books, list the numbers of *all* the questions answered. *Do not write your name on any answer book.*
3. Return all answer books in the folder provided. Additional answer books are available if needed.

SPECIFIC INSTRUCTIONS:

Answer:

- either one (1) of the questions G540-1 or G540-2, *and*
- both (2) questions in the section labeled B760, B761, B766, or B780 corresponding to your chosen focus area, *and*
- any two (2) additional question in the sections B760, B761, B766, and B780, where these two questions need *not* come from the same section, *and*
- both (2) questions in the section labeled A7xx that corresponds to your focus area.

Hence, you are to answer a total of *seven* (7) questions.

POLICY ON MISPRINTS AND AMBIGUITIES:

The Exam Committee tries to proofread the exam as carefully as possible. Nevertheless, the exam sometimes contains misprints and ambiguities. If you are convinced that a problem has been stated incorrectly, mention this to the proctor. If necessary, the proctor can contact a representative of the area to resolve problems during the *first hour* of the exam. In any case, you should indicate your interpretation of the problem in your written answer. Your interpretation should be such that the problem is nontrivial.

NOTE:

This is the last semester that questions will be given related to the course CS 761.

Answer either one (1) of the questions G540-1 or G540-2.

G540 – Introduction to AI Questions

G540-1. Represent the following using the *situation calculus*:

> Robby the robot can move a block B that is on the table to the top of another block C, provided both blocks currently have nothing on top of them.

Write and explain one relevant *frame axiom* for use with the above *move* action.

G540-2. Suppose you are trying to solve the following puzzle using the *A* search algorithm*. The puzzle involves numbers from 100 to 999. You're given two numbers called S and G. You're also given a set of numbers called *Bad*. A move consists of transforming one number into another by adding 1 to one of its digits, or subtracting 1 from one of its digits. For example, a move can take you from 678 to 679, or from 234 to 134. All moves have cost 1. Moves are subject to the following constraints:

- You cannot add to the digit 9 or subtract from the digit 0.
- You cannot make a move that transforms your current number into one of the numbers in the set *Bad*.
- You cannot change the same digit twice in two successive moves.

You solve the puzzle by getting from S to G in the fewest number of moves.

(a) Describe a *state description* that is sufficient for this problem and can be used with the A* algorithm.

(b) Define an *admissible heuristic* for use with A* search in this problem. Explain why your heuristic is admissible.

(c) Use your heuristic in (b) to carry out the first three (3) node expansions of A* search when $S = 567$ and $G = 777$, and $Bad = (666, 667)$. Include in your tree all legal successors of each node you expand.

Answer both (2) of the questions in the section labeled B7xx that corresponds to your chosen focus area. Also answer any two (2) additional questions in any of the other sections (these two questions need NOT occur in the same section).

B760 – Machine Learning Basic Questions

B760-1. Assume you wish to create, using human-labeled training examples, a system that separates interesting images from uninteresting ones (where each user defines interestingness according to his or her personal interests). Sketch how you could use the algorithms (a) *Naive Bayes* and (b) *k-Nearest Neighbors* to accomplish this. Be sure to briefly explain how you might represent images, any assumptions built into these two ML algorithms, and how you'd select good parameter settings (e.g., k).

B760-2. *Overfitting* is a well known, central problem in machine learning. Briefly explain this issue, how it arises in decision-tree induction, and one method for addressing it.

Given that overfitting is a problem, presumably *underfitting* is as well. Can underfitting occur in ID3/C4.5? If so, how might this problem be addressed?

Finally, when using Q-tables, can overfitting occur in reinforcement learning in a noise-free, deterministic environment? Explain why or why not.

B761 – Deductive Problem Solving & Logic Programming Basic Questions

B761-1. Convert the following logical sentences to clauses, and then derive the empty clause using binary resolution plus paramodulation.

1. $\forall x (x = x)$.
2. $\forall x \forall y \big(cuz(x, y) \;\Leftrightarrow\; (x \neq y \;\wedge\; \exists u \exists z (p(u, x) \wedge p(z, y) \wedge sib(u, z)))\big)$.
3. $\forall x \forall y \big(sib(x, y) \;\Leftrightarrow\; (x \neq y \;\wedge\; \exists u (p(u, x) \wedge p(u, y)))\big)$.
4. $\neg \forall xy \big(cuz(x, y) \;\Rightarrow\; cuz(y, x)\big)$.

You may also use hyper-resolution steps if you want.

B761-2. Consider the Prolog database:

$honest(X) \;:-\; \neg liar(X).$
$liar(X) \;:-\; says(X, Y), lie(Y).$
$says(a, p).$
$says(b, q).$
$says(b, r).$
$lie(r).$

Here, $\neg$ is Prolog's negation, written "`not`" in some prologs and "`\+`" in other prologs. What does Prolog answer to the queries $?- \, honest(a)$, $?- \, honest(b)$, and $?- \, honest(c)$? Explain your answer, both in terms of Prolog's search mechanism, and in terms of the negation-as-failure semantics.

B766 – Computer Vision Basic Questions

B766-1. *Edge Detection*

- *(a)* Describe two major differences between the Canny edge detector and the Marr-Hildreth (i.e., Laplacian-of-Gaussian) edge detector.
- *(b)* The statement is often made that "edges occur at different scales." What does this mean? Give an example to illustrate.
- *(c)* How can one deal in principle with the "scale" problem? Give specific steps for the case where we are interested in finding good object structure and not constituent texture. (I.e., which scales would you consider, and how would you put the results from different scales together?)

B766-2. *Correspondence Problem*

- *(a)* Define the *correspondence problem* as it occurs in stereo vision.
- *(b)* Describe a region-based approach for solving the correspondence problem using *cross-correlation*.
- *(c)* Edge-based approaches for solving the correspondence problem often use additional constraints such as the *monotonicity constraint*, which assumes that matching points along epipolar lines have the same order in each image. Prove whether or not this constraint holds in general.

B780 – Robot Motion Planning Basic Questions

B780-1. A mobile robot knows its own coordinates and its goal coordinates at all times, and has a rangefinder sensor that allows it to sense nearby obstacles. Does the following *sensor-based motion planning algorithm converge*?: go toward the goal; if meet an obstacle, pass it on the left until you can go again toward the goal; iterate. Prove your response.

B780-2. Consider a *3-link RRP arm manipulator* (R="revolute," P="prismatic"), that behaves like the so-called Scara arm (i.e., the first two links rotate in the horizontal plane, and the third slides up and down vertically). Does the inverse kinematics for this arm have a unique solution? If not, how many? Draw a sketch; explain your answer in detail.

Answer both (2) of the questions in the section labeled A7xx that corresponds to your chosen focus area.

A766 – Computer Vision Advanced Questions

A766-1. Say we want to *recognize 2D objects* based on point features (corresponding, say, to corners). Given an object model, we must recognize instances of the object in any position, orientation and scale (i.e., 4 degrees of freedom) in an image.

(a) Describe the main steps in either the *alignment method* or the *geometric hashing method* for solving this problem.

(b) Characterize how the method you chose in (a) could be interpreted as a search in "correspondence space" (i.e., correspondences between image features and model features) and/or "pose space" (i.e., possible transformations of a model).

(c) How can, or why can't, the method you chose in (a) deal with (i) a large database of models, and (ii) partial occlusion?

A766-2. Suppose you are given a *video* of the outside of the State Capitol. Describe some of the key problems in *constructing a 3D model* of the building using only the images in the video. Also describe the main steps in an algorithm for doing this reconstruction.