EXAMINATIONS — 2002

END-YEAR (corrected)

COMP 307

ARTIFICIAL INTELLIGENCE

(corrected)

Time Allowed: 3 Hours

Instructions: There are a total of 180 marks on this exam.
Attempt all questions.
Calculators may be used.
Non-electronic foreign language translation dictionaries may be used.

Questions

1. Prolog [25]
2. Search [20]
3. State Space Problem solving [20]
4. Rule Based Systems [20]
5. Machine Learning [25]
6. Planning [25]
7. Natural Language Processing. [25]
8. Artificial Intelligence and Chess. [20]

Question 1. Prolog [25 marks]

(a) [6 marks] For each of the queries below, state the answer that prolog would give ("yes", "no", or a variable binding).

For example,

```
|?- a = a.                  yes
|?- 4 = 2+3.                no
|?- [a, b] = [X, b].        X=a
```

```
|?- X=isa(c1, cup).

|?- X = 5+3.

|?- X is 5+3.

|?- X=5, X is X+3.

|?- first \== second.

|?- att(cup1, X)= att(Y, white).

|?- att(cup1, X)= att(X, white).

|?- [cup, bench, kitchen] = [X | Y].

|?- [c1, on(Y, Z), in(b, k)] = [X, on(X, b) | W].
```

(Question 1 continued on next page)

Consider the following prolog program.

```
in(p0, kitchen).
in(c1, diningroom).
in(c2, kitchen).
in(c3, kitchen).
in(c4, kitchen).

plate(p0).
cup(c1).
cup(c2).
cup(c3).
cup(c4).

clean(p0).
dirty(c1).
clean(c2).
dirty(c3).
clean(c4).

toWash(X):-in(X, kitchen), format("~n In: ~w",    [X]),
           cup(X),         format("~n Cup: ~w",   [X]),
           dirty(X),       format("~n Dirty: ~w",[X]),
           fail.
```

(b) [6 marks] What is the output of the following query?

```
|?- toWash(X).
```

(Question 1 continued on next page)

(c) [3 marks] Suppose a cut were added to the `toWash` predicate (on the second line):

```
toWash(X):-in(X, kitchen), format("~n In: ~w",   [X]),
           cup(X),         format("~n Cup: ~w",  [X]), !,
           dirty(X),       format("~n Dirty: ~w",[X]), fail.
```

What would the output of the following query be?

```
|?- toWash(X).
```

(d) [10 marks] .
Write a predicate, `relevantTo(X, L, L2)`, where `L` is a list of terms and `L2` is a sublist of `L` containing just the terms that have `X` as their first argument. For example,

```
|?- relevantTo(c1, [on(c1, bench), on(c2, table), china(c1),
                isa(c2, cup), att(c1, colour, white)], Z).

Z = [on(c1, bench), china(c1), att(c1, colour, white)]
```

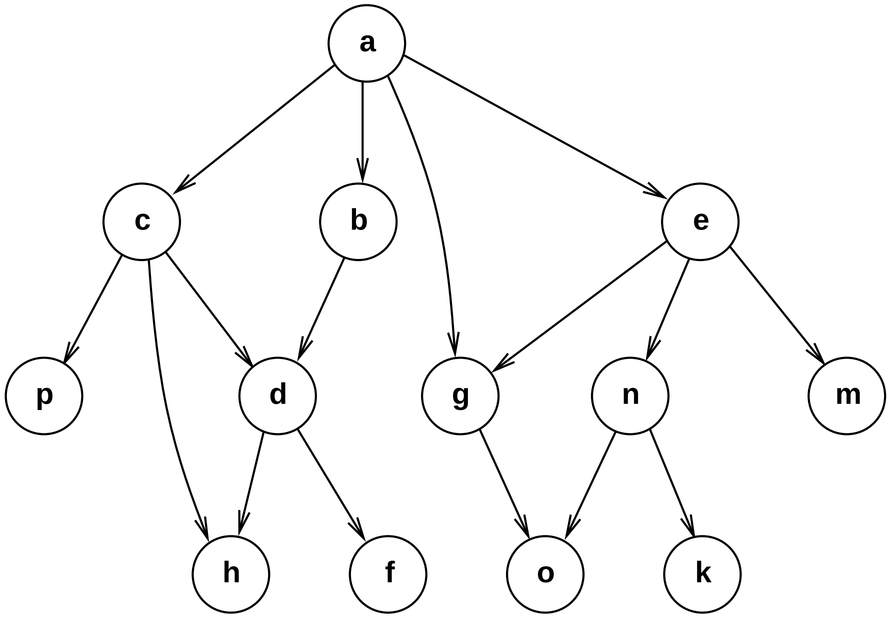
You will need to use the operator `=..` that "deconstructs" a term into a list of the head and arguments of the term:

```
|?- in(c2,kitchen) =.. X.
X = [in, c2, kitchen]
```

```
relevantTo(
```

Question 2. Search [20 marks]

(a) [8 marks] Consider the following state space, where each node represents a state, and each directed link represents an operation.



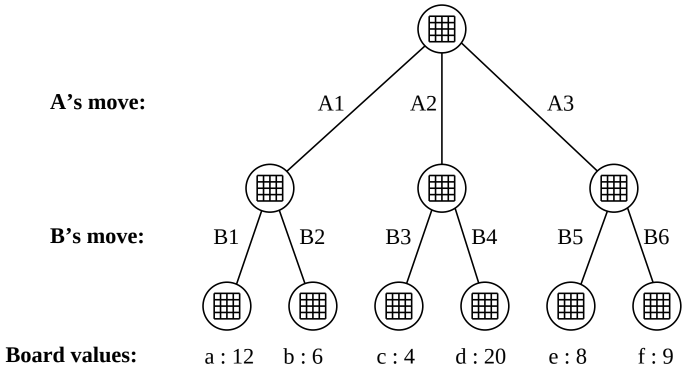
Suppose the state space is to be searched, starting at node "a". List the nodes of the state space in the order that they are expanded for each of the following search methods. Assume that the search considers child nodes from left to right. State any other assumptions you make.

Depth First Search:

Breadth First Search:

Iterative Deepening:

Consider the following tree of board positions from a two person board game like chess. The tree shows that player A has three possible moves from the current board position and player B has two possible moves from each of the resulting positions. The values of the board positions after each of B's moves are shown at the bottom (larger numbers are better for A).



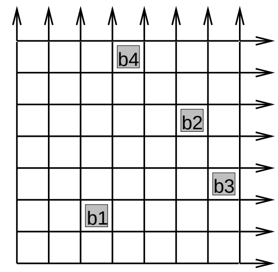
(b) [3 marks] Which move should player A make, and why?

(c) [3 marks] If evaluating board positions is expensive, we want to avoid evaluating any board unless necessary. Which board position(s) from "a" to "f" do not need to be evaluated to determine the best move for A? Explain why.

(d) [6 marks] Depth First, Hill Climbing, and A* search use different stopping criteria. State the stopping criteria for each kind of search and briefly indicate what class of search problem they are appropriate for.

Question 3. State Space Problem Solving [20 marks]

Consider a robot in a world consisting of a collection of labeled boxes that can each be placed in any square of an infinite grid (at most one block per grid square). At each step, the robot can push one box by one step in any direction (N, S, E, W).



A state in the state space can be specified by giving a location (x,y) for each box (assume four boxes):

`state(b1(3,2), b2(6,5), b3(7,3), b4(4,7))`

The possible actions could be specified by the box and the direction:

`move(b1, N), move(b1, S), move(b1, E), move(b1, W)`
`move(b2, N), move(b2, S),` *etc*

If the robot were given initial and final states:

initial: `state(b1(3,2), b2(6,5), b3(7,3), b4(4,7))`
final: `state(b1(13,2), b2(6,15), b3(7,1), b4(1,1))`

the robot could search for a sequence of moves to get from the initial state to the final state.

(a) [3 marks] Show the first three states that the robot might consider if it used depth first search:

(b) [3 marks] Why would depth first search be a very bad strategy for the robot?

(Question 3 continued on next page)

(c) [3 marks] Breadth first search would be a better strategy. Why is it still not a good strategy?

(d) [3 marks] Why would iterative deepening be a better strategy than breadth first?

(e) [8 marks] Suggest a heuristic that would make A* search a reasonable search strategy for this problem domain. Explain why your heuristic is admissible.

Question 4. Rule Based Systems [20 marks]

Consider the following set of rules for a rule based system for determining whether there is oil at a location on the basis of various geological features.

1.	**If** alkalite **and** high watertable	**then** fold morphology
2.	**If** quartz	**then** fold morphology
3.	**If** shale **and** crystallite	**then** slope morphology
4.	**If** schist **and** alkalite	**then** slope morphology
5.	**If** fold morphology **and** low chryolation	**then** deep oil
6.	**If** fold morphology **and** calcification	**then** shallow oil
7.	**If** slope morphology **and** moraine-transfer	**then** deep oil
8.	**If** slope morphology **and** sinkage	**then** no oil

Suppose we were to use a backward chaining inference system that cached answers and intermediate conclusions and considered the three hypotheses in order:

shallow oil, deep oil, no oil

Suppose also that when asked about a fact, the user would give the following answers:
(note, the true facts are also underlined in the rules above)

A.	alkalite	no
B.	calcification	no
C.	low chryolation	no
D.	crystallite	yes
E.	moraine-transfer	yes
F.	quartz	yes
G.	schist	yes
H.	shale	yes
I.	sinkage	no
J.	high watertable	no

(a) [5 marks] Show the sequence of rules the inference engine would consider.

(b) [4 marks] Show the sequence of facts it would ask about.

(Question 4 continued on next page)

(c) [1 mark] State which hypothesis it would conclude.

(d) [4 marks] What explanation might the system give if asked to justify its conclusion?

(e) [6 marks] A rule based system such as the one above could not diagnose the presence or absence of oil with certainty. State at least three different sources of uncertainty and suggest a simple modification to the rules to incorporate this uncertainty.

Question 5. Machine Learning [25 marks]

(a) [5 marks] Briefly outline the Decision Tree building algorithm that constructs a decision tree given a set of instances each labeled with their class.

(b) [4 marks] State the desirable properties for an “impurity measure” on a set of instances, and give an example of an acceptable impurity measure.

(c) [6 marks] Consider the following data set describing 10 kinds of mushrooms of which 5 are poisonous and 5 are safe. They are described by three attributes.

Colour	**Stalk**	**Top**	**Class**
red	tapered	flat	Poisonous
red	tapered	round	Poisonous
red	bulbous	round	Poisonous
brown	bulbous	round	Poisonous
brown	bulbous	round	Poisonous
brown	tapered	round	Safe
brown	tapered	round	Safe
white	tapered	round	Safe
white	bulbous	round	Safe
white	bulbous	flat	Safe

(Question 5 continued on next page)

(Question 5 continued)

Which attribute would the decision tree building algorithm choose for the root of a decision tree for the mushroom data? Show your working and state which impurity measure you are using.

(d) [5 marks] A perceptron is a linear weighted threshold device. It has several (numeric) inputs and a boolean output. If the input values of a perceptron are represented by the variables $x_1, x_2, \ldots x_k$, give the formula for computing the output of the perceptron. Explain any variables you use other than x_i.

(e) [5 marks] Very briefly outline the algorithm for learning the weights of a perceptron from a set of examples. You do not have to specify exactly how the weights are modified.

Question 6. Planning

[25 marks]

(a) [4 marks] What are the four parts of a STRIPS operator?

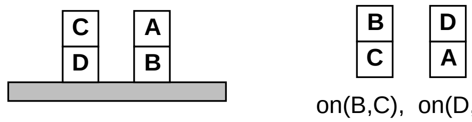
(b) [4 marks] Write a STRIPS operator for the action `paint`. To paint an object a particular colour, the robot must be holding a paint brush of the right colour. The effect of the action is to change the colour of the object.

```
paint(
```

(c) [4 marks] STRIPS operators are very limited in the kinds of actions that they can represent. List at least four properties of actions for real world robots that cannot be easily represented in standard STRIPS operators,

(Question 6 continued on next page)

(d) [6 marks] Explain why a goal-stack planner would have difficulty finding a good solution to the following blocks world problem? (Assume the "standard" operators of pickup, putdown, stack, and unstack.)



on(B,C), on(D, A)

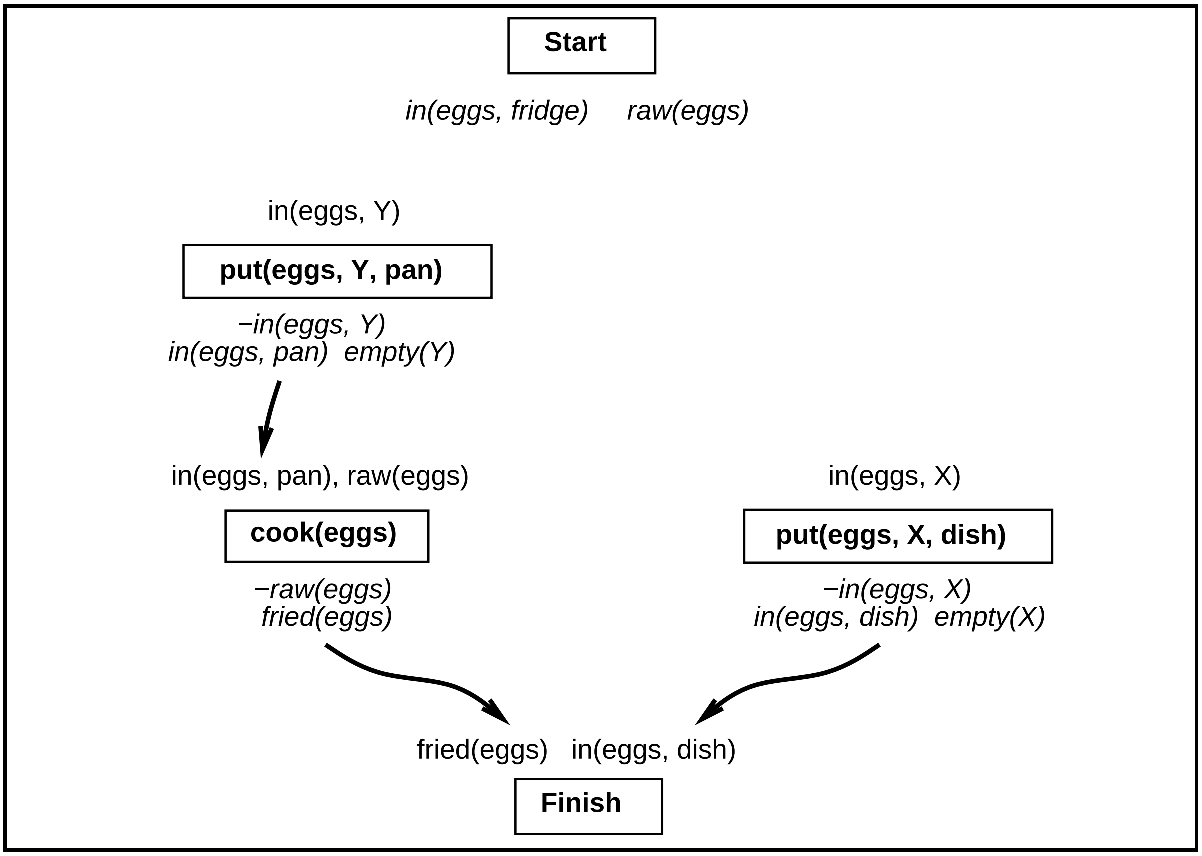
Initial state **Goal State**

(Question 6 continued on next page)

Suppose we are using a Partial Order Planner (POP) to construct a plan for a meal of fried eggs in a dish. The diagram below shows the state of the plan after the planner has added three new actions to satisfy some of the preconditions.

(e) [4 marks] The last action added has introduced two threats into the plan. Identify the threats on the plan and show two different ways to resolve them.

(f) [3 marks] Show (on the diagram) two possible ways of satisfying the "in(eggs, X)" precondition of the `put(eggs, X, dish)` action on the right, and indicate which one of them will lead to a plan that works.



SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

Question 7. Natural Language Processing [25 marks]

Consider the following grammar and lexicon for simple robot commands such as

"put the plate in the kitchen":

```
s --> verb, np, pp.

pp --> prep, np.

np --> det, np1.

np1 --> adj, np1.
np1 --> noun, np1.
np1 --> noun.

prep --> [P], {lexicon(P, prep)}.
det  --> [D], {lexicon(D, det)}.
noun --> [N], {lexicon(N, noun)}.
verb --> [V], {lexicon(V, verb)}.
adj  --> [A], {lexicon(A, adj)}.
```

```
lexicon(cup, noun).
lexicon(plate, noun).
lexicon(bench, noun).
lexicon(kitchen, noun).

lexicon(take, verb).
lexicon(get, verb).
lexicon(put, verb).
lexicon(place, verb).

lexicon(white, adj).
lexicon(blue, adj).

lexicon(to, prep).
lexicon(from, prep).
lexicon(on, prep).
lexicon(in, prep).

lexicon(the, det).
lexicon(a, det).
```

(a) [4 marks] Show the parse tree for the command

"take the plate to the kitchen"

(Question 7 continued on next page)

(b) [3 marks] Show the parse tree for the command

"put a white cup on the kitchen bench"

(c) [3 marks] Why can't the grammar parse the sentence

"place the blue plate on the cup in the kitchen"

(d) [4 marks] Replace the rule for `np` by a rule or rules that let the grammar also correctly parse the sentence

"place the blue plate on the cup in the kitchen"

```
np -->
```

(Question 7 continued on next page)

(e) [11 marks] The original grammar is too general – it allows invalid sentences, like the two on the left below, although the two on the right are valid.

"*take the cup on the kitchen"	"take the cup to the kitchen"
"*put the plate to the bench"	"put the plate on the bench"

The problem is that the preposition in the PP doesn't match the verb:

- The verbs "take" and "get" must have a PP with a location-specifying preposition: "to" or "from".
- The verbs "put" and "place" must have a PP with an object-specifying preposition: "on" or "in".

Extend the grammar and the lexicon to enforce the constraint that the preposition matches the verb. You will need to modify some rules and the lexicon, but you do not need to add any new grammar rules. You do not need to modify the lexicon entries for nouns, adjectives, or determiners.

```
s          --> verb       , np       , pp.

pp         --> prep       , np.

np         --> det        , np1.
np1        --> adj        , np1.
np1        --> noun       , np1.
np1        --> noun.
prep        --> [P], lexicon(P, prep).
det         --> [D], lexicon(D, det).
noun        --> [N], lexicon(N, noun).
verb        --> [V], lexicon(V, verb).
adj         --> [A], lexicon(A, adj).

{lexicon(take, verb)}.
{lexicon(get, verb)}.
{lexicon(put, verb)}.
{lexicon(place, verb)}.

{lexicon(to, prep)}.
{lexicon(from,  prep)}.
{lexicon(on, prep)}.
{lexicon(in, prep)}.
```

Question 8. Artificial Intelligence and Chess [20 marks]

Making a computer that could play chess well was one of the early goals of Artificial Intelligence researchers. Computers can now play chess at grand master level, but it is not clear whether this involves intelligence. Explain why you think chess computers such as Deep Blue do, or do not, involve any intelligence, and whether creating chess computers contributes to our understanding of artificial intelligence.

****