EXAMINATIONS — 2001

End-of-year

COMP 307

Introduction to Artificial Intelligence

Time Allowed: 3 Hours

Instructions: Attempt ALL Questions.

The exam will be marked out of 180.
Non-programmable calculators without full alphabet keys are permitted.
Non-electronic foreign language dictionaries are permitted.

Questions

	Marks
1. Prolog	[32]
2. Search	[32]
3. Knowledge-based Systems	[23]
4. Machine Learning	[30]
5. Planning	[33]
6. Natural Language Processing	[30]

Question 1. Prolog [32 marks]

(a) [12 marks] How Prolog works

Consider the query:

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

Give the output for each of the following programs.

Note: You must give all the possible answers. If an answer is found more than once, show all the occurrences. If the answer is "no", very breifly explain why.

(i)

```
a(X):-b(X), c(X), d(X).
b(1).
c(2).
d(3).
```

(ii)

```
a(X):-b(X), c(X), d(X).
b(1).
b(2).
c(1).
c(2).
d(2).
```

(iii)

```
a(X):-b(X), c(X).
a(X):-d(X).
b(1).
b(2).
c(1).
c(2).
d(2).
```

(iv)

```
a(X):-b(X), !, c(X).
a(X):-d(X).
b(1).
b(2).
c(1).
c(2).
d(2).
```

(b) [10 marks] List and Recursion

Consider the following program.

```
append([], L, L):-
      print_list(L),
      nl.                     % nl prints a new line
append([H|T], L, [H|LT]):-
      append(T, L, LT),
      print_list([H|LT]),
      nl.

print_list([]).
print_list([H|T]):-
  write(H),
  print_list(T).
```

What is the output of the program for the following query?

```
|?- append([a, b, c], [1, 2, 3, 4], X).
```

You must give all the messages printed to the screen and the output for the variable binding.

(c) [10 marks] Lists

Define a predicate `min_number(NumberList, Min)`, which finds the smallest number in a list of numbers. For example

```
|?- min_number([2, 7, 1, 4], X).
X=1.
```

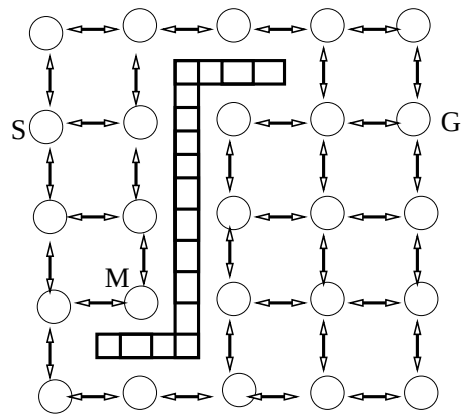
SPARE PAGE FOR EXTRA ANSWERS

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

Question 2. Search

[32 marks]

Consider the following 5×5 grid environment, where S is the start state, G is the goal state and M is a middle state (M is only used in question (e)). The arrows show possible movements between states, and there is a "wall" blocking some transitions. There are a maximum of four operators that can be applied to any state: `up, down, left, right`. Suppose the cost of an operator is always one.



(a) [4 marks] Describe an appropriate Prolog representation for a state in this search problem. Show the initial state S and the goal state G under your representation.

(b) [4 marks] Describe a way to represent the "wall" and give an example(in Prolog).

(c) [6 marks] Define Prolog predicate `action(State1, MoveMent, State2)` to represent the **up** operator. You must consider the conditions that makes the **up** operator legal.

(d) [8 marks] Draw a fragment of the search tree, showing all the states within two steps of the initial state. Suppose the four operators are applied in this order: `up, down, left, right`.

(e) [10 marks] A* search uses an evaluation function $f(n) = g(n) + h(n)$.

Consider a general search problem,

- Explain the two components of the function.
- What are the constraints on $g(n)$ and $h(n)$?

Consider the search problem in the 5×5 grid environment,

- Give a function that can be used as $h(n)$ for this problem.
- For the initial state S and the state M as shown in the 5×5 grid environment, what is the value of $g(S), h(S), g(M), h(M)$?

Question 3. Knowledge-based Systems [23 marks]

(a) [8 marks] Rule-based Systems

Consider the following set of sentences:

- A student is successful if he/she has high grades.
- Students who are bright and work hard have high grades.
- Students who are not bright do not pass comp101.
- Students who do not work hard have lots of fun.
- Mary is not having any fun.
- Mary passes comp101.

Translate the sentences into rules and facts in Prolog. Your program should be able to conclude that Mary is successful (You do not need to show the reasoning process). You must choose the appropriate vocabulary and design for your predicates. Do not include student as a predicate, as it complicates the solution.

(b) [10 marks] Semantic Networks

Draw a single semantic network to represent the following statements:

> E1 is an elephant and its name is Clyde. Elephants are grey and each elephant has 1 tail, 1 trunk, and 4 legs.

(c) [5 marks] Knowledge Representation

Consider the following sentences:

> When you go to the cinema, you usually buy a ticket, hand the ticket to the ticket taker and go and find a seat. Sometimes you buy popcorn before going to your seat. When the movie is over, you leave the theater.
>
> John went to see a movie.

What would be an appropriate knowledge representation formalism for these sentences, that would facilitate answering the question:

> Did John buy a ticket?

Briefly explain how the formalism would be helpful to answer the question.

Question 4. Machine Learning [30 marks]

(a) [10 marks] Learning Decision Trees

We have data about 10 soccer games and we want to learn a decision tree to predict "win" or "loss". The training set is as follows:

Week	Location	Weather	Own Record	Opposition Record	Result
1	home	hot	good	good	win
2	home	rain	good	average	win
3	away	hot	good	poor	win
4	away	cold	poor	average	win
5	home	hot	poor	poor	win
6	home	moderate	good	average	win
7	away	moderate	good	average	loss
8	home	cold	good	good	loss
9	away	hot	average	average	loss
10	home	moderate	average	good	loss

Table 1: the training set

If we use the decision tree building algorithm presented in lectures, which attribute would be chosen for **the root** of the decision tree? Show your working.

(b) [10 marks] Neural Networks

Using the data given in Table 1, suppose we want to train a neural network to learn the weights to predict “win” or “loss”.

Describe the architecture of a neural network for this learning task. Do not worry about the hidden layer, just give the number of input nodes, the number of output nodes, and the encoding of the input nodes and the output nodes (e.g. what does “1” or “0” means for each node).

Consider the first two training examples shown in Table 1, rewrite them using your input and output encoding.

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.

(c) [10 marks] Genetic Algorithm

Using the data given in Table 1, suppose we want to use a genetic algorithm to learn the rules to predict "win"or "loss". Consider these rules:

> If location is home, weather is hot, own record is good, opposition record is poor, then win.
>
> If own record is good (other attributes can have any values), then win.

Describe a way to represent rules so that they can be used as individuals in a genetic algorithm. Rewrite these two rules using your representation.

What would be a suitable fitness function for this problem? For the rule "if own record is good then win", what is the evaluation value using your fitness function?

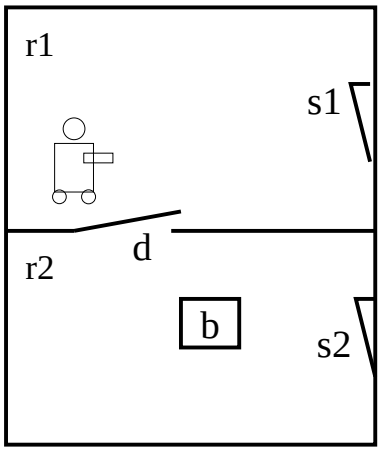
Question 5. Planning [33 marks]

Shakey the robot lives in a world consisting of two rooms `r1 & r2`, one door `d`, one box `b` and two light switches `s1 & s2`. The door connects the two rooms. Each room has a light switch.

There are many possible actions. Shakey can

- walk to any location in a room (door `d`, switch `s1`(or `s2`), boxLocation, etc).
- open or close a door.
- move from one room to another.
- push a box to any location in the same room.
- carry a box from one room to another (there is a step at the door and Shakey can not push a box through).
- climb up onto and get down from a box.
- turn a light switch on and off.

Let us focus on the `turnOn(Switch)` action. Since Shakey is short, he can only turn on a light switch if he is on top of a box and he and the box are both at the switch position, in the same room.



(a) [8 marks] Define the `turnOn(s1)` action (turn on the light switch in room 1) using the format we introduced in State Space Search. The action should be represented as: `action(state1, movement, State2)`. The state representation should be expressive enough to represent any states.

(b) [8 marks] Define the `turnOn(s1)` action in STRIPS.

(c) [9 marks] Define the `turnOn(s1)` action in situation calculus, including all the effect axioms and frame axioms that are related to the `turnOn(s1)` action.

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.

(d) [8 marks] Assuming Shakey uses least commitment planning, draw the plan it will generate for the following task.

> At the initial state, Shakey is in room 1, the door is closed, the box is in room 2 and all light switches are turned off.
>
> The goal state is that the light switch in room 1 (`s1`) is turned on and the door is closed.

You only need to draw the final plan and you may omit the preconditions and the effects of the operators.

Start

Finish

Question 6. Natural Language Processing [30 marks]

(a) [15 marks] DCG Parser

Consider the following DCG parser.

```
sentence(s(NP, VP))-->np(NP), vp(VP).

np(np(NP1, pp(PP)))-->np1(NP1), pp(PP).
np(NP)-->np1(NP).

np1(np(det(D), n(N)))-->det(D), n(N).

vp(vp(v(V), adv(Ad)))-->v(V), adv(Ad).
vp(vp(v(V)))-->v(V).

n(students)-->[students].
det(the)-->[the].
v(worked)-->[worked].
adv(hard)-->[hard].
pp([in, vuw])-->[in, vuw].
```

This parser can recognize a small set of English sentences and generate the parse trees.

(i) [3 marks] What is the output for the following query? If the answer is no, briefly explain why.

```
|?-sentence(Tree, [the, students, worked], []).
```

(ii) [6 marks] What is the output for the following query? If the answer is no, briefly explain why.

```
|?-sentence(Tree, [the, students, in, vuw, worked, hard], []).
```

(iii) [6 marks] What is the output for the following query? If the answer is no, briefly explain why.

```
|?-sentence(Tree, [the, students, worked, hard, in, vuw], []).
```

(b) [15 marks] Write a DCG program to translate prices in English into numbers. Here are some examples:

Prices in English	Value as a real number
twenty five dollars sixty five cents	25.65
twenty dollars sixty cents	20.6
thirteen dollars sixty eight cents	13.68
nine dollars twenty cents	9.2
three dollars five cents	3.05
eight dollars	8
one dollar fifty five cents	1.55
one dollar	1
eighty cents	0.8
one cent	0.01

We assume the price is always lower than $100. The lexicon is given as follows:

```
% for 1~9
digit(1)-->[1].
digit(2)-->[2].
...

% for 10~19
teen(10)--[ten].
teen(11)-->[eleven].
teen(12)-->[twelve].
...

% for 20~90
ten(20)-->[twenty].
ten(30)-->[thirty].
...
```

We also defined `d, c` as follows:

```
d-->[dollars].
d-->[dollar].
c-->[cents].
c-->[cent].
```

Write DCG rules that can recognize a price in English and translate it into a real number. You may start with the general cases and then add special rules for special cases.

****

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.

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.

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.

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.