THE UNIVERSITY OF AUCKLAND

FIRST SEMESTER, 2003
Campus: City

COMPUTER SCIENCE

Foundations of Artificial Intelligence

(Time allowed: ONE AND A HALF hours)

NOTE: Closed book.
No calculators.
Attempt all questions.
Put the answers in the boxes below the questions.

MARKS	
HANS:	(out of 20)
MIKE:	(out of 40)
PAT:	(out of 40)
TOTAL:	(out of 100)

SURNAME:

FORENAME(S):

STUDENT ID:

UNIVERSITY UPI:

UNIVERSITY UPI: ____________________

Section A: Prolog

1. Consider the following Prolog program:

```
next_oldest(anton,bert). next_oldest(bert,carmen).
next_oldest(carmen,diane). next_oldest(diane,emil).
next_oldest(emil,fiona). next_oldest(fiona,greg).
older(X,Y) :- next_oldest(X,Y).
older(X,Z) :- next_oldest(Y,Z), older(X,Y).
```

(a) What would be Prolog's answer to the query `next_oldest(carmen,fiona)`. [1 mark]

no

(b) What would be Prolog's answer to the query `older(anton,bert)`. [1 mark]

yes

(c) List Prolog's answer to the query `older(X,diane)` in the order that they are produced. [2 marks]

X = carmen, X = bert, X = anton, no

(d) List Prolog's answer to the query `older(emil,Y)` in the order that they are produced. [2 marks]

Y = fiona, Y = greg, no

2. What is the first answer of the following Prolog program when queried with a constant as first argument, a list as second argument, and a variable as third argument? [3 marks]

```
mystery(X,[],[]).
mystery(X,[X|L1],[X,X|L2]) :- mystery(X,L1,L2).
mystery(X,[Y|L1],[Y|L2]) :- mystery(X,L1,L2).
```

The second argument with every occurence of the first argument duplicated.

UNIVERSITY UPI: ____________________

Section B: Search

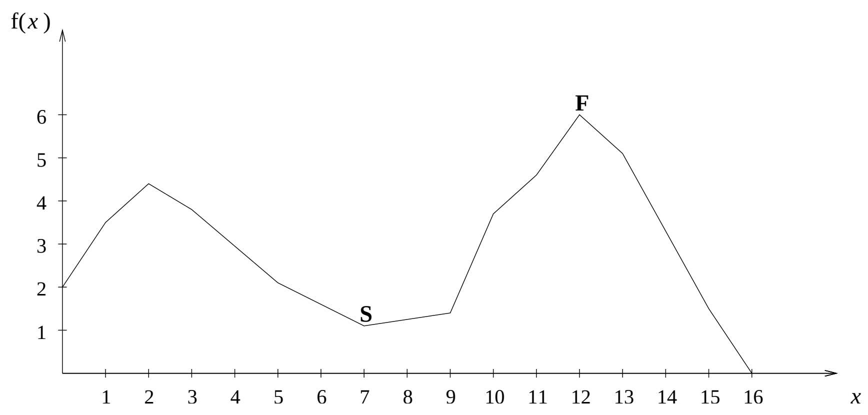
3. Tick the search strategy which is more straightforward to implement in Prolog. [2 marks]

Breadth-first search	☐	Depth-first search	√

4. Each of the following graphs shows a one-dimensional search space in which the states are represented by integers. The quality of a state x is denoted by $\mathrm{f}(x)$). Each state x has two successor states: $x - 1$ and $x + 1$. The final state is marked by an **F**. Mark that state with an **S** for which the following holds:

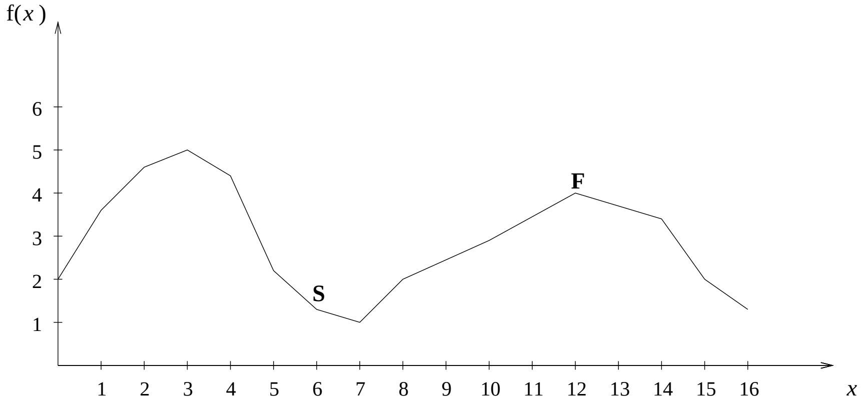
- Steepest-ascent hill climbing applied to **S** would *not* result in **F**.
- **S** is the state closest to **F** with that property.

(a) [2 marks]

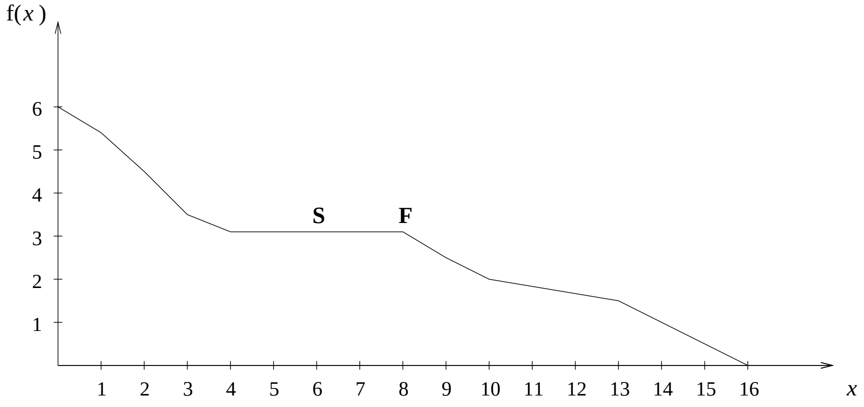


UNIVERSITY UPI: ______________________________

(b) [2 marks]



(c) [2 marks]



UNIVERSITY UPI: ____________________

5. Which of the following statements are false?

   (a) Since uniform cost search minimizes the cost of the path from the start to the current node, it always finds an optimal solution.

   (b) A* search minimizes the estimated cost of the path from the start through the current node to the goal.

   (c) Since hill climbing always selects a state better than the current state, it terminates in a global optimum.

   (d) Since greedy search minimizes the estimated cost of the path from the current node to the goal, it always finds an optimal solution.

   (e) Simulated annealing avoids getting stuck in a local optimum by accepting a worse state from time to time.

[3 marks]

(c), (d)

UNIVERSITY UPI: ______________________________

Section C: Planning

6. World States: Closed World versus Open World Assumptions [6 marks]

Assume we represent world states as lists of literals, e.g., [on(a, table), not(clear(a)), ...].

(a) Closed World Assumption [3 marks]

Assume we represent world states using the closed world assumption.

i. Assume we are told that neither clear(b) nor not(clear(b)) are in the list of literals that describe the current state of the world, what does this tell us about what we believe about whether b is clear in the current state?

we believe b is not clear in the current state

ii. If we want to assert that ”b is clear” is not true in the current state, how would that be represented in our list of literals?

by making sure that clear(b) is not in the list

iii. If we want to assert that we don’t know whether ”b is clear” is true in the current state, how would that be represented in our list of literals?

the closed world assumption does not allow us to represent this

(b) Open World Assumption [3 marks]

Assume we represent world states using the open world assumption.

i. Assume we are told that neither clear(b) nor not(clear(b)) are in the list of literals that describe the current state of the world, what does this tell us about what we believe about whether b is clear in the current state?

it tells us that we have no beliefs about whether b is clear or not in the current state

UNIVERSITY UPI: ______________________________

ii. If we want to assert that ”b is clear” is not true in the current state, how would that be represented in our list of literals?

we make sure that clear(b) is not is the list and that not(clear(b)) is in the list

iii. If we want to assert that we don’t know whether ”b is clear” is true in the current state, how would that be represented in our list of literals?

we make sure that neither clear(b) nor not(clear(b)) are in the list

7. Iterative Deepening Search [4 marks]

Assume you have the following search space:

```
        a
       / \
      b   c
     / \  |
    d   e f
   / \   / \
  g   h i   j
  |  / \
  k l   m
```

Where only node h is a solution.

(a) How many times does node k get visited?

0 times

(b) How many times does node c get visited?

2 times

UNIVERSITY UPI: ____________________

8. Progression versus Regression Planning [8 marks]

Assume we only have the following domain actions:

```
moveToTable(Block, From)
   preconditions: [on(Block, From), clear(Block), isBlock(From)]
   effects: [not(on(Block, From)), on(Block, table), clear(From)]

moveFromTable(Block, To)
   preconditions: [on(Block, table), isBlock(Block),
                   clear(To), clear(Block)]
   effects: [not(on(Block, table)), on(Block, To), not(clear(To))]
```

and the following problem description:

```
initial state: [on(a,b), on(b,table), clear(a), on(c,table), clear(c)]
goal description: [on(a,c)].
```

(a) Progression Planner [4 marks]

Assume we are using a progression planner.

i. Assume our current plan is simply the empty plan, how many different steps might we add to plan at this point (i.e., there are a number of different steps we might choose from, to add at this point, how many alternative steps are there to choose from)?

2 alternatives

ii. What are they?

they are:
- moveToTable(a,b)
- moveFromTable(c,a)

(b) Regression Planner [4 marks]

Assume we are using a regression planner.

i. Assume our current plan is simply the empty plan, how many different steps might we add to plan at this point (i.e., there are a number of different steps we might choose from, to add at this point, how many alternative steps are there to choose from)?

1 alternative

UNIVERSITY UPI: ______________________________

ii. What are they?

moveFromTable(a,c)

9. Goal Regression [8 marks]

Assume we have the following domain actions:

```
moveToTable(Block, From)
   preconditions: [on(Block, From), clear(Block), isBlock(From)]
   effects: [not(on(Block, From)), on(Block, table), clear(From)]

moveFromTable(Block, To)
   preconditions: [on(Block, table), isBlock(Block),
                   clear(To), clear(Block)]
   effects: [not(on(Block, table)), on(Block, To), not(clear(To))]
```

and the following plan

```
[step(moveToTable, [a,b]),
 step(moveToTable, [b,c])],
```

where the steps are in the order they are to be executed.

What would be regressed goal sets if we regressed the goal clear(c) through that entire plan? Show both the final regressed goal set and the intermediate regressed goal set (make it clear which is which).

regressing clear(c) through moveToTable(b,c) gives the intermediate goal set [on(b,c), clear(b), isBlock(c)] and regressing that through moveToTable(a,b) gives the goal set [on(b,c), on(a,b), clear(a), isBlock(b), isBlock(c)]

UNIVERSITY UPI: ______________________________

10. Object-level and meta-level predicates [4 marks]

(a) Action preconditions can contain both object-level and meta-level conditions. What is the difference between them?

> Object-level conditions are those whose truth value is determined by looking at the world state while meta-level conditions' truth value cannot be so determined.

(b) Give an example of each from the blocksWorld.

> example of object-level condition - on(a,b)
> example of meta-level condition - neq(A,c)

UNIVERSITY UPI: ______________________

11. Satisfaction [10 marks]

Assume the closed world assumption, that world states are represented by a list of positive object-level literals, and that goals are represented by a list of both positive and negative literals (do not worry about handling the neq/2 literal (e.g., neq(A, B)).

Write the prolog code for satisfiedBy(GoalList, WorldState) which is true if and only if each goal in GoalList is true in WorldState.

```
satisfiedBy([], _).

satisfiedBy([not(P) | Rest], State) :-
      !,
      not(member(P, State)),
      satisfiedBy(Rest, State).

satisfiedBy([P | Rest], State) :-
      member(P, State),
      satisfiedBy(Rest, State).
```

UNIVERSITY UPI: ______________________

Section D: Logic

12. What is the difference between entailment and inference? [2 marks]

If A entails B then B must be true in all worlds where A is true. If A infers B that means there is a procedure that can prove B from A.

13. Draw the sets of models for A and KB, assuming KB $\models$ A [2 marks]

The picture will be a venn diagram where the set KB is completely inside the set A

14. Which would you rather have a procedure that is sound but not complete, or a procedure that is complete but not sound? You must explain why! [2 marks]

A sound procedure is better than a complete one because you need to be able to trust that the answers are in fact true.

UNIVERSITY UPI: ______________________________

15. If you were making a agent to crawl the web, which do you think would cause you more problems with this agent, the frame problem, the qualification problem, or the ramification problem? You must explain why! [2 marks]

    The frame problem is more important then the other two because you can't have a system that functions without handling the frame problem. The other two are less critical to the systems success.

16. Why is entailemnt only semidecidable for first order logic?? [2 marks]

    Entailment will return an answer if it is valid or unsatisfiable but if it is only satisfiable it will not be able to determine an answer.

17. Which rule is diagnostic and which rule is causal? [1 mark]

    (a) $Worried(X, P) \rightarrow Didn'tStudy(x, P)$

        diagnostic

    (b) $Didn'tStudy(X, P) \rightarrow Worried(x, P)$

        causal

UNIVERSITY UPI: ______________________________

18. Prove $PassedTest(Pat, 366)$ using the following axioms: [15 marks]

$StudiedHard(x) \land NotTired(x) \rightarrow PassedTest(x, 366)$
$StudiedHard(Pat)$
$NoParties(x) \land NoAllNighter(x) \rightarrow NotTired(x)$
$NoParties(Pat)$
$NoAllNighter(Pat)$

(a) by forward chaining

The answer must go forward from axioms to goal.

(b) by backward chaining

The answer must go backward from goal to axioms.

UNIVERSITY UPI: ______________________________

(c) by resolution

The answer must include negating the goal and deriving an empty set.

19. Write an English sentence for each of these formulas: [4 marks]

(a) $\forall x likes(x, brocolli)$

Everyone likes brocolli.

(b) $\neg\forall x \neg likes(x, brocolli)$

Someone likes brocolli.

(c) $\forall x \exists y likes(x, y)$

Everyone likes someone.

(d) $\exists x \forall y likes(x, y)$

Someone likes everyone.

UNIVERSITY UPI: ____________________

20. Unify the following sets of axioms if possible. a and b are literials and x,y,z are variables. [5 marks]

(a) $\{R(b), R(f(x))\}$

FAIL

(b) $\{P(a, y), P(x, f(y))\}$

FAIL

(c) $\{S(a, x), S(y, f(y))\}$

$\{a/y, f(a)/x\}$

(d) $\{P(x), P(y)\}$

$\{y/x\}$

(e) $\{S(x, y), S(f(a), a)\}$

$\{f(a)/x, a/y\}$

UNIVERSITY UPI: ______________________

21. Put the following statements into clausal form. [5 marks]

(a) $\exists x \forall y \exists u : (R(x) \rightarrow [S(y) \wedge Q(u)])$

$\neg R(a) \vee S(y_1)$
$\neg R(a) \vee Q(f(y_2))$

(b) $\forall u \forall y : ([S(u) \vee Q(y)] \rightarrow \exists x : R(x, y, u))$

$\neg S(u) \vee R(f(u, y), y, u))$
$\neg Q(w) \vee R(f(v, w), w, v))$

(c) $\forall x\{S(x) \rightarrow \exists y\{R(x, y)\}\} \wedge \forall x\{\neg S(x) \rightarrow \neg \exists y\{Q(y, x)\}\}$

$\neg S(x) \vee R(x, f(x))$
$S(u) \vee \neg Q(w, u)$