THE UNIVERSITY OF AUCKLAND

SECOND SEMESTER, 2015
Campus: City

COMPSCI.367
The Practice of Artificial Intelligence

(Time allowed: 60 minutes)

This test is out of **100** marks.
Attempt **ALL** questions.
Write your answers in the space provided in this booklet. There is space at the back for answers that overflow the allotted space.
The use of calculators is **NOT** permitted.

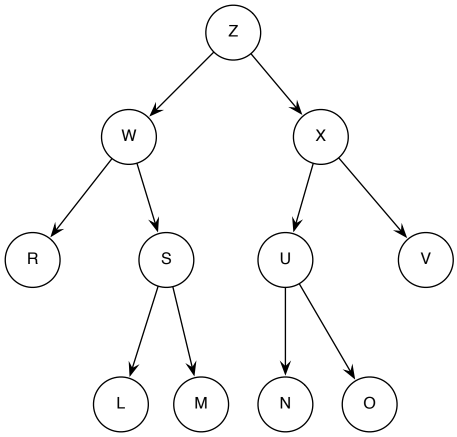
Surname (Family Name):	
First Name(s):	
UoA ID Number:	
Login Name (UPI):	

Section	Mark	Marks Available
A		47
B		53
Total		100

Print Name:

PART A

Question 1



Say in which order the nodes are expanded using three different search methodologies? Assume that the entire tree is expanded? [6 marks]

a) breadth first search

Z,W,X,R,S,U,V,L,M,N,O

b) depth first search

Z,W,R,S,L,M,X,U,N,O,V

c) iterative deepening

Z;Z,W,X;Z,W,R,S,X,U,V;Z,W,R,S,L,M,X,U,N,O,V

Print Name:

Question 2

Which of the following algorithms are complete and optimal? If you make assumptions, please specify them. [6 marks]

a) breadth first search

complete and optimal if the branching factor is not infinite

b) depth first search

not optimal but complete if the depth is not infinite and there are no loops

c) iterative deepening

complete and optimal if branching factor is not infinite

Question 3

With the following knowledge base, which of the following sentences are entailed? [6 marks]

KB = A and ((B and C) or D) and (E and (F or G))

a) A

True

b) B

False

c) E

True

Print Name:

Question 4

Given the following statements and trying to prove Q, what order will the conjuncts be touched using forward chaining? Draw the graph if that will help you answer the question. [3 marks]

G => Q
C ^ E => G
Z ^ C => E
X ^ G => C
X ^ Z => C
X
Z

forward chaining

X,Z,C,E,G,Q

Question 5

Why is simulated annealing used more often than the other techniques for real world problems?

[6 marks]

It is used in a lot of situations because it can move to a worse state to escape a local optima. This is very useful if you do not know whether you have local optima in your space.

Print Name:

Question 6

What are two of the ways we discussed in class to improve the results from Hill Climbing? Please explain why they improve Hill Climbing, to get full marks [6 marks]

Random restarts – pick a random state, do hill climbing, when stuck stop, repeat X times and return best answer found

Sideways moves – can move sideways if no uphill moves, must stop after X consecutive sideways moves could make an infinite loop

Stochastic – choose randomly among the uphill moves instead of always picking the best

First-choice – type of stochastic, pick first uphill move expanded and don't expand any more – this is especially good if each state has many uphill sucessors

Question 7

What is overcrowding in genetic algorithms? Of the main parts of an genetic algorithm: crossover, mutation, selection, and fitness function, which two affect overcrowding the most and why? Please explain your answer for full marks. [6 marks]

Overcrowding is when one hypothesis in the population has much higher fitness than the rest of the population and quickly takes over the population causing early convergence at a local optima. Selection and the fitness function have the biggest impact on overcrowding. Fitness proportionate/roulette wheel selection will increase overcrowding. Rank or tournament selection will reduce overcrowding. A fitness function with large flat areas and a single tall thin spike or with lots of tall spikes will increase overcrowding, while a fitness function like rolling hills will decrease overcrowding.

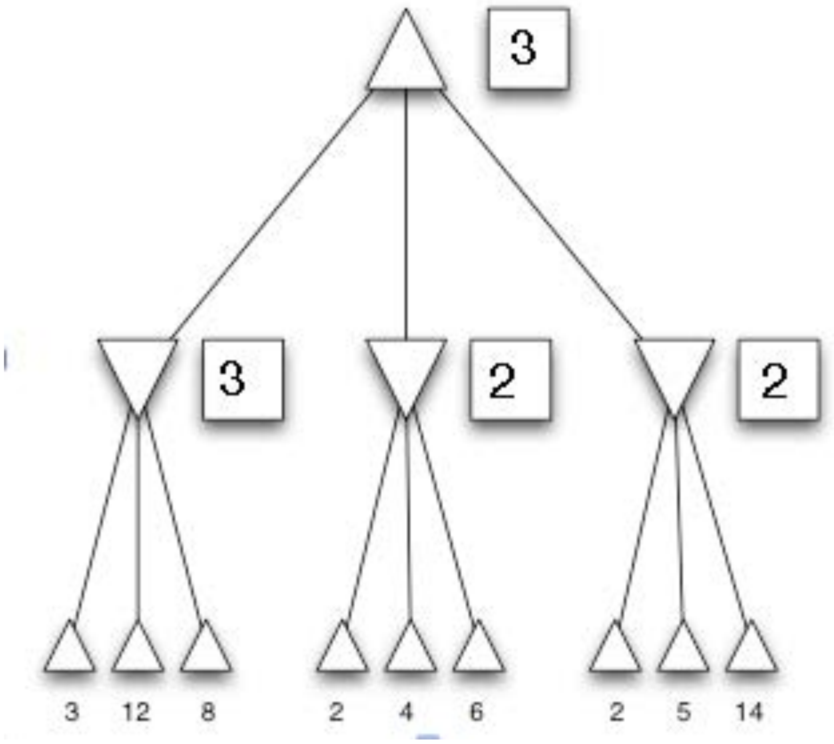
Continued…

Print Name:

Question 8

Given the following game tree, mark the values for each node in the box provided. Assume you are using minimax.

[4 marks]

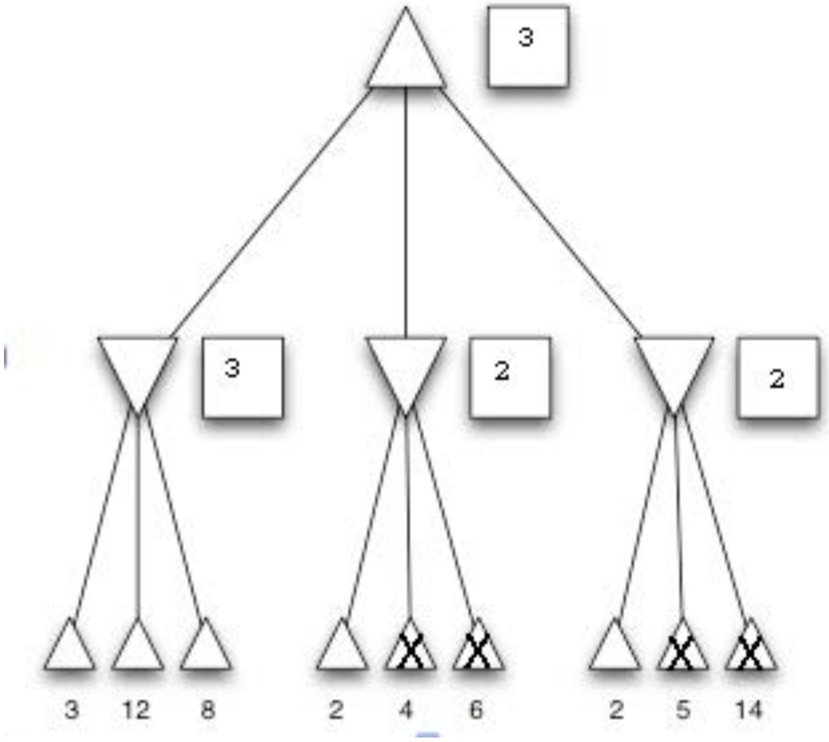


Print Name:

Question 9

Given the following game tree, mark the values for each node in the box provided. Assume you are using minimax with alpha-beta pruning (make sure you make clear which nodes are not expanded). Also make clear which values are updated as the algorithm runs.

[4 marks]



Print Name:

PART B:

Question 10

Translate the following first-order logic expression into English. [5 marks]

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

there exists someone who everyone likes

Question 11

Translate the following English expression into first-order logic. [5 marks]

Everybody has somebody they like.

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

Question 12

Which of the following sentences will unify? If they do unify, give their substitution list, if they don't unify, write "fail" under substitution list. [10 marks]

		Substitution List
A(John, y)	A(x, Pat)	{x/John, y/Pat}
A(P(x), x)	A(P(John), y)	*{x/John, y/John}*
A(Q(x,y))	A(z)	*{z/Q(x,y)}*
A(P(x)))	A(x)	*fail*
A(P(x), y)	A(z,x)	*{z/P(x), y/x}* *{z/P(x), x/y}*
A(P(x), Q(y))	A(u,u)	*fail*

Continued…

Print Name:

Question 13

Translate the following first-order logic into Prolog. [5 marks]

$\forall x\ (\exists y\ ((\text{Hardworking}(x) \lor \text{Goodlooking}(x)) \land \text{LooksAfter}(y,x)) \rightarrow \text{Happy}(x))$

happy(X) :- (hardworking(X) ; goodlooking(X)), looksAfter(Y, X).

or

happy(X) :- hardworking(X), looksAfter(Y,X).
happy(X) :- goodlooking(X), looksAfter(Y,X).

Question 14

Translate the following Prolog into first-order logic. [5 marks]

isStack(X) :- emptyStack(X) ; (popStack(X, Top, Rest), isStack(Rest)).

$\forall x\ (\text{EmptyStack}(x) \lor (\text{PopStack}(x, top, rest) \land \text{IsStack}(rest)\ )\ ) \rightarrow \text{IsStack}(x)$

Print Name:

Question 15

Give an advantage and a disadvantage for each of the following search methods, for full marks make it clear that you understand what these advantages/disadvantages are, not just what they are called. [6 marks]

Greedy Heuristic Search

Adv: might find a solution quickly

Disadv: no guarantees on (optimality, completeness)

A*

Adv: guaranteed (optimal solution, complete)

Disadv: requires memory exponential wrt solution length

IDA*

Adv: requires memory linear wrt solution length

Disadv:cannot do full duplicate state elimination/pruning, so can expand exponentially more nodes since it cannot detect when it has already expanded a node in a different part of the search tree

Question 16

What are the benefits for **A*** if its heuristic is (a) admissible, and (b) consistent? [6 marks]

(a)

if the heuristic is admissible then the first time A expands a goal node, we are guaranteed that its path is optimal*

(b)

if the heuristic is consistent then the first time A expands a node with a specific state, s, we are guaranteed that the path to s is optimal*

Print Name:

Question 17

Do admissible consistent heuristics always reduce the time **A*** takes to solve a problem? Explain why. [5 marks]

While using an admissible consistent heuristic will never cause A to expand more nodes, it may not reduce the time A* takes to solve the problem. This is because the time to solve the problem depends both upon the number of nodes expanded and on the time it expand a node. The node expansion time includes the time it takes A* to evaluate the new nodes with the heuristic. So for a given problem, the heuristic may reduce the number of nodes in the search tree it has also increased the amount of time taken to expand each node. Thus, if the size of search space is only slightly reduced but the time to expand a node is greatly increased, A* may end taking longer to solve the problem with the heuristic than it would taken without the heuristic.*

Print Name:

Question 18 [6 marks]

(a) When a goal is a complete state description, how would bidirectional seach detect when the two frontiers collide, what would its "cost" be?

If the goal is a complete state description and assuming that the backward direction search also produces complete state description then bidirectional search could detect the collision of the two frontiers by using hash tables for both directions' closed lists and when we choose a node from one direction's open list we could check for collision by hashing the state into the other direction's closed list.

Assuming an appropriately sized hash table and a reasonable hash function the cost for this test would be O(1) with respect to the length of the solution path length.

(b) When the goal is only a partial state description, how could bidirectional search detect when the two frontiers collide, what would its "cost" be?

If the goal is only a partial state description and assuming that the backward search produces successive different partial state descriptions then using hashing to detect collisions will not work. Instead, if we choose a forward search node to expand, we could check for a collision by checking whether its state satsifes any of the partial state descriptions in the backwards direction closed list, and if we choose a backwards search node to expand, we can check for collision by checking whether any of states in the forward search's closed satisfies our chosen node's goal description.

Assuming that the test for satisfiability of a goal by a state is a constant time cost, then the cost for the collision test would be exponential with respect to the solution path length.

This page left blank for any questions that overflow.

Print Name:

This page left blank for any questions that overflow.