CS 416, Artificial Intelligence
Midterm Examination

Fall 2004

Name:___________________________

This is a closed book, closed note exam. All questions and subquestions are equally weighted.

Introductory Material

1) **True or False**: In this course, we are studying rational agents and what it means to think rationally. FALSE… *ACT* Rationally

2) Provide a synonym for stochastic: <u>Random, Markov, probabilistic, non-deterministic</u>

Uninformed Search

3) When comparing tree-search algorithms, we measure the number of nodes expanded. How many nodes are expanded (in the worst case) by each of the following search techniques when searching a tree with branching factor *b* to find a goal at a depth of *d*? You can uses ellipses in your answer to indicate a sequence. Do not use big Oh notation.

a) Breadth-first search: $b+b^2+b^3+ \ldots + b^d + b^{d+1} - b$

b) Depth-first search: infinity

c) Depth-limited search (limit = *d*): $b + b^2 + \ldots + b^d$

d) Iterative deepening depth-first search: $db + (d-1)b^2 + \ldots b^d$

Informed Search

4) What is the definition of an admissible heuristic?

Never over estimates the cost to the goal.

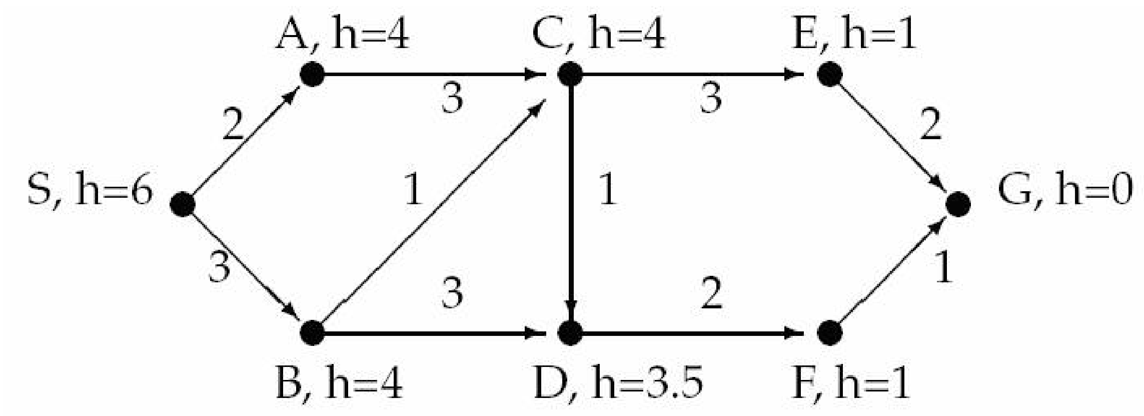
5) **True or False**: The greedy best-first search algorithm is complete.

FALSE: infinite loops are possible

6) What complication regarding completeness can be caused when augmenting A* to include the provision for deleting revisited states?

1) potential loss of optimality, but not completeness
2) complicated to make sure the right one is deleted
3) shorter paths are possible and longer paths are possible
4) thrashing

7) Suppose we want to use the A* algorithm on the graph below to find the shortest path from node S to node G. Each node is labeled by a capital letter and the value of a heuristic function. Each edge is labeled by the cost to traverse that edge.



For this problem:

- Perform the A* algorithm on this graph, filling in the table below. You should not need all the lines in the table. Indicate the *f*, *g*, and *h* values of each node on the queue as shown in the first two rows of the table. You need not write the contents of the (priority) queue in order in the table.

  Assume that if you find a path to a node already on the queue that you update its cost (using the lower *f* value) instead of adding another copy of that node to the queue.

- Show the path found by the A* algorithm on the graph above.

iteration	node expanded	Priority queue at end of this iteration
0		S = 0 + 6 = 6 (i.e. S = g(S) + h(S) = f(S))
1	S	A = 2 + 4 = 6; B = 3 + 4 = 7
2	A	C=9, B=7
3	B	C=8, D=9.5
4	C	D=8.5, E=8
5	E	G=9, D=8.5
6	D	F=8, G=9
7	F	G=8
8	G	

b) Make this example inadmissible by changing the heuristic value at one of the nodes. What node do you choose and what heuristic value do you assign?

node ____E______ Note the problem already was inadmissible at D

heuristic value ______3______

c) Is A* using Tree-Search algorithm still complete? YES

d) How would the search be affected by your change?

Not optimal

8) Iterative deepening A* (IDA*) uses the cost function (g + h) to determine how much further to explore the search space (as opposed to iterative deepening depth-first search which used the depth of the tree). How much does IDA* increment the search cutoff after each iteration?

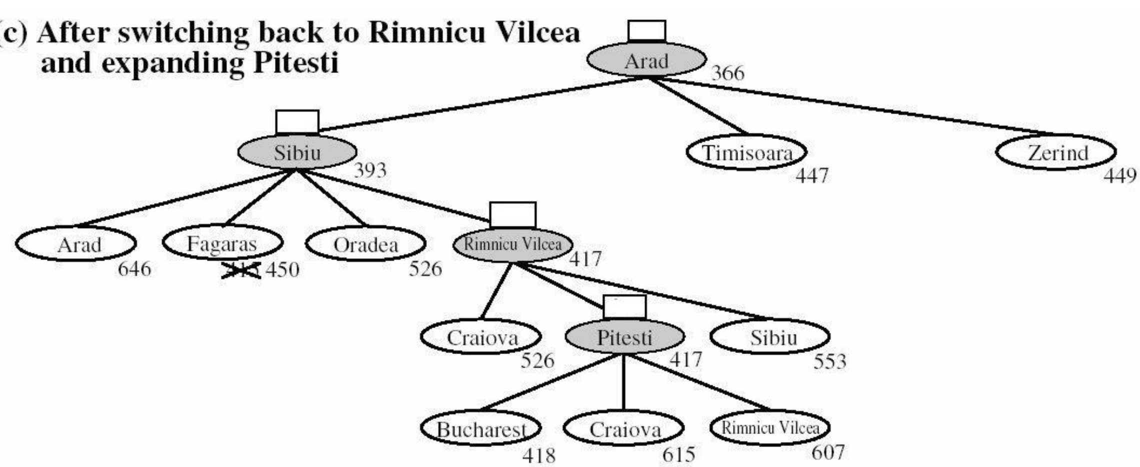
Cutoff is updated to the smallest f-cost of all nodes that exceeded cutoff in the previous iteration

9) What is an example of when IDA* becomes inefficient?

Small increments in f-cost as with real-valued evaluation functions. Having a deep path to the solution is not enough to cause IDA* to be particularly inefficient. Any search will have to incur the cost of reaching the depth of the solution.

10) The search tree below was generated by recursive best-first search after switching back to Rimnicu Vilcea and expanding Pitesti. The f-value is missing from four rectangles on the search tree. **Add them.** infinity at ARAD. 447 at Sibiu. 447 at Rimnicu Vilcea, 447 at Petesti.

(c) After switching back to Rimnicu Vilcea and expanding Pitesti



11) What is an example of a *relaxed problem* for the 15-puzzle?

Fewer tiles, Manhattan distance, misplaced tiles, slide tiles over others

12) What is a shortcoming of hill climbing algorithms?

Local mins and plateaus

13) In each iteration of simulated annealing, only one move is evaluated.

a) Specifically, how is that one chosen? At random

b) **True or False:** The evaluated move is always discarded if it is worse than the current state and another move is chose. FALSE

c) **True or False:** It is possible to get stuck in a local maximum in simulated annealing. TRUE

14) Newton-Raphson is used to find what feature of an equation? How is this useful in searching for optimal solutions?

We use Newton-Raphson to find the root of an equation (where $f(x) = 0$). By finding the roots of $f'(x)$ – the derivative – we find the local mins and maxes.

Adversarial Games

15) In the following minimax tree, alpha/beta pruning was utilized. Answer the following questions:

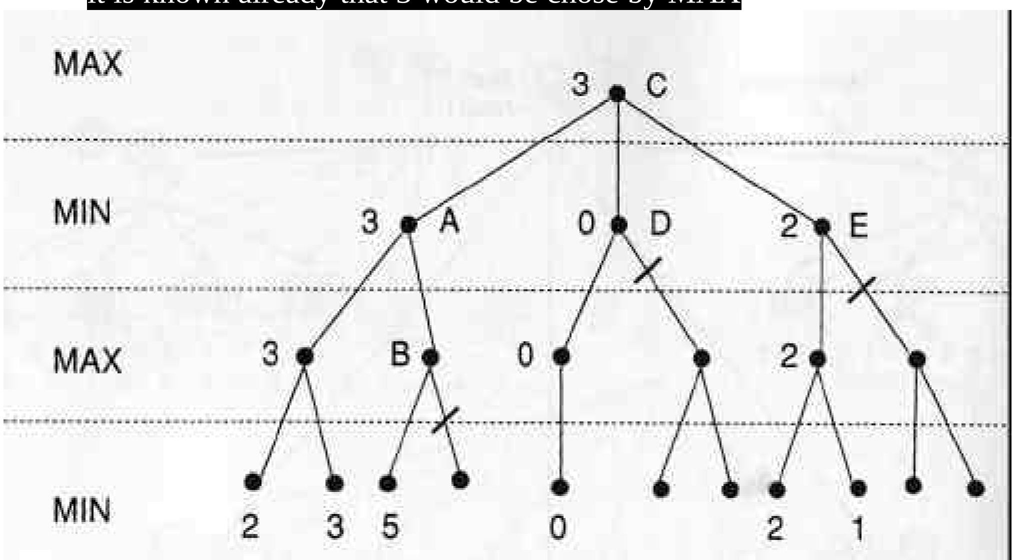
a) The value at node A indicates the outcome will be (**no larger / no smaller**) than 3.

b) Node B is pruned because: even if the outcome were larger than 5 and thus desired by MAX, MIN at the level above would always pick 3 to minimize damage

c) The value at node C indicates the outcome will be (**no larger / no smaller**) than 3.

d) Node D is pruned because: MIN cannot beat a score of 0

e) Node E is pruned because: While it is possible MIN could find a score less than 2, it is known already that 3 would be chose by MAX



16) In the figure above, assume that one and only one node can be a chance node. That is, the successor to that node is determined by a coin toss, not by the utility-maximizing choices of a player. In this version of the game, assume the minimum value of an outcome is zero.

a) Describe how the minimax exploration of the game would change if node A were the chance node. MIN cannot control outcome there. Outcome would be 3 or at least 5.

Would the value at node C change? Yes, E(½ * 3 + ½ 5 (or something better))

b) Describe how the minimax exploration of the game would change if node D were the chance node. MIN cannot control outcome. Outcome would be 0 or something arbitrarily large. MAX might choose the new outcome > 0.

Would the value at node C change?

Maybe, it depends on the leaf nodes below D (> 3)

Propositional Logic

17) An inference algorithm is sound if it: Does not infer false statements

18) An inference algorithm is complete if it: Derives any sentence that is entailed

19) A sentence is (valid **or satisfiable**) if it is true in *all* models.

20) α entails β if and only if (____ $a \wedge \neg b$ ______________) is unsatisfiable.

21) In the space provided to the right of the knowledge base (KB) printed below, rewrite the KB in conjunctive normal form.

$V \vee T$	V ∨ T
$P \wedge \neg U$	~P ∨ U
$R \vee \neg Q$	R ∨ ~Q
$V \Rightarrow W$	~V ∨ W
$P \Rightarrow Q$	~P ∨ Q
$S \Rightarrow (U \vee T)$	~S ∨ U ∨ T
$(P \wedge R) \Rightarrow S$	~(P∧R) ∨ S = ~P ∨ ~R ∨ S

b) In this KB, show an example of how And-Elimination can be used to add a new sentence to the KB.

$$\frac{V \wedge T}{V}, \frac{V \wedge T}{T}$$

c) In this KB, show an example of how Modus Ponens can be used to add a new sentence to the KB.

$$\frac{V \Rightarrow W, V}{W}$$

d) In the CNF KB from above, show an example of how *resolution* can be used to add a new sentence to the KB.

$$\frac{R \vee \neg Q, \quad \neg P \vee \neg R \vee S}{\neg Q \vee \neg P \vee S}$$

e) If you were to use resolution to demonstrate the above KB entails T, what sentence would you add to the KB to start the process? $\neg T$

f) Use you answer to the previous question and the KB to show the proposition T is entailed (or provide a proof that it is not entailed). Use only the inference rules of propositional logic.

~S ∨ U ∨ T	∧	~T	
~S ∨ U	∧	~U (AND-ELIM)	
~S	∧	~P ∨ ~R ∨ S	
~P ∨ ~R	∧	P (AND-ELIM)	
~R	∧	R ∨ ~Q	
~Q	∧	~P ∨ Q	
~P	∧	P (AND-ELIM)	FALSE

First-Order Logic

22) What single inference rule is complete when partnered with a complete search algorithm?

Resolution (also accepted Modus Ponens)

23) Represent the following sentences in first order logic:

a) All dogs are mortal $\forall \boldsymbol{x}(\boldsymbol{Dog}(\boldsymbol{x}) \Rightarrow \boldsymbol{Mortal}(\boldsymbol{x}))$

b) No person buys an expensive policy $\neg\exists \boldsymbol{x}(\boldsymbol{Person}(\boldsymbol{x}) \wedge \boldsymbol{BuysExpensivePolicy}(\boldsymbol{x}))$

24) **True or False:** Diagnostic rules lead from observed effects to hidden causes?

25) In the following two examples, skolemize to eliminate the existential quantifier:

$$\exists\, x \;\; Crown(x) \wedge OnHead(x, John)$$

$$\forall\, x \;\; [\exists\, y \;\; Animal(y) \wedge \neg Loves(x, y)] \vee [\exists\, z \;\; Loves(z, x)]$$

Crown (C_1) ∧ OnHead (C_1, John)

$\forall \boldsymbol{x}[\boldsymbol{Animal}(\boldsymbol{F}_1(\boldsymbol{x})) \wedge \neg \boldsymbol{Loves}(\boldsymbol{x}, \boldsymbol{F}_1(\boldsymbol{x})] \vee \boldsymbol{Loves}(\boldsymbol{F}_2(\boldsymbol{x}), \boldsymbol{x})$

26) In the following two examples, provide a unification if one exists. Label all cases where you standardize sentences apart in order to complete the unification:

Knows (John, x) and Knows (x, Elizabeth)
Standardize apart: Knows (John, x) and Knows (y, Elixabeth)
{x/Elizabeth, y/John)

Knows (John, x) and Knows (y, Mother(y)) {x/Mother(John), y/John}

Older (Father (y), y) and Older (Father(x), John) {x/John, y/John}

27) Every sentence of first-order logic can be converted into an inferentially equivalent sentence of what form? Accepted lots of answers because the question wasn't sufficiently precise… BNF, Propositional Logic, CNF, Horn Clauses, Predicate Calculus

28) Is it **forward chaining or backward chaining** that runs a greater risk of not being a complete search technique in first-order logic?

29) What does it mean to say that entailment for first-order logic is semidecidable?

Algorithms exist that return YES to every entailed sentence, but no algorithm exists that also returns NO to every nonentailed sentence.