CS 540-1: Introduction to Artificial Intelligence

Exam 1: 7:15-9:15pm, March 2, 1998

CLOSED BOOK

(one page of notes and a calculator allowed)

Write your answers on these pages and show your work. If you feel that a question is not fully specified, state any assumptions you need to make in order to solve the problem. You may use the backs of these sheets for scratch work.

Write your name on this and all other pages of this exam. Make sure your exam contains six problems on seven pages.

Name ____

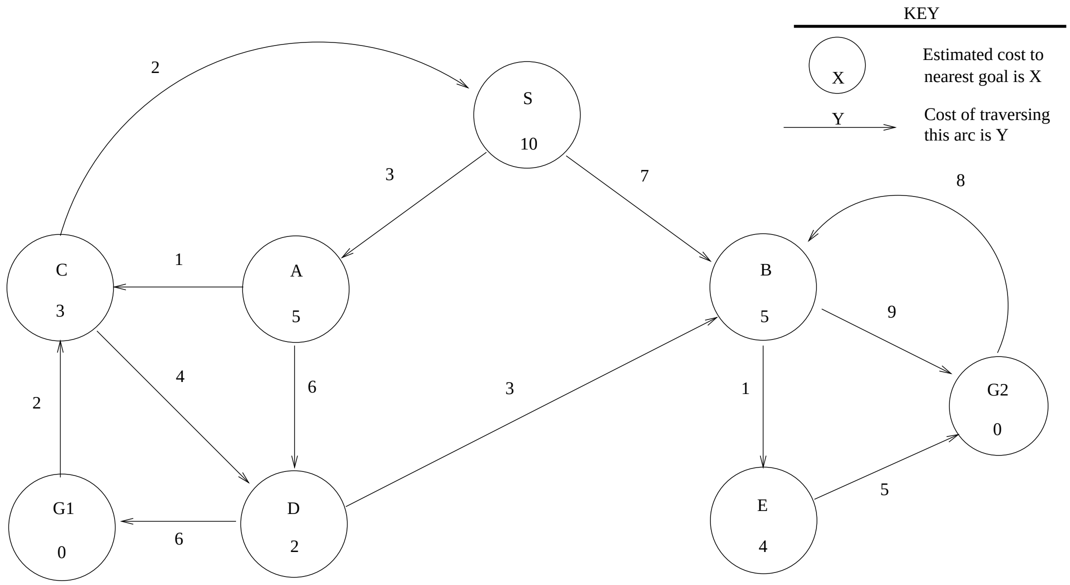
Student ID ____

Problem	**Score**	**Max Score**
1	_____	20
2	_____	10
3	_____	10
4	_____	10
5	_____	40
6	_____	10
Total	_____	100

(over)

PROBLEM 1 - Basic Search Strategies (20 points)

Assume you have the following search graph, where *S* is the start node and *G1* and *G2* are goal nodes. Arcs are labeled with the cost of traversing them and the estimated cost to a goal is reported inside nodes.



For each of the search strategies listed below, indicate which goal state is reached (if any) and list, in order, the states expanded. (Recall that a state is *expanded* when it is *removed* from the OPEN list.) *When all else is equal, nodes should be expanded in alphabetical order.*

Depth-First

Goal state reached: _____ States expanded: ____

Iterative Deepening

Goal state reached: _____ States expanded: ____

Hill Climbing (using the *h* function only)

Goal state reached: _____ States expanded: ____

A*

Goal state reached: _____ States expanded: ____

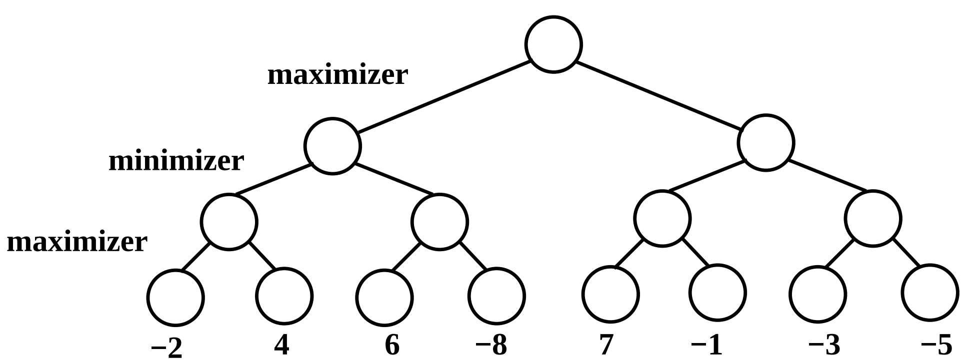
PROBLEM 2 - Heuristic Functions (10 points)

Imagine you are trying to solve the following puzzle using the *A** search algorithm. The puzzle involves two-digit numbers from 10 to 99. There are two legal actions: you can add 1 to any digit except 9 and you can subtract 1 from any digit except 0. (E.g., 10 can be changed, in one step, to 20 and 11, while 99 can be changed to 89 and 98). There is one restriction: you can never create the number 55. All actions have cost 100. The task is to find the shortest solution path from a given *start* number to a given *goal* number.

Design and justify an *admissible* heuristic function for this puzzle.

PROBLEM 3 - Game Playing (10 points)

Apply the *minimax* algorithm to the game tree below, where it is the *maximizer's* turn to play. The values estimated by the static-board evaluator (SBE) are indicated below the leaf nodes. Write the estimated values of the intermediate nodes inside their circles, and indicate the proper move of the maximizer by circling one of the root's outgoing arcs.



Indicate, by crossing out, *one* (1) unnecessary call to the static-board evaluator. Explain why this call to the board evaluator is unnecessary.

PROBLEM 4 - Miscellaneous Short Answers (10 points)

Provide brief answers to the following questions.

A. Why are *pruning sets* used in decision-tree algorithms?

B. If a solution exists, is *depth-first search* guaranteed to find it?

C. Can *iterative deepening* ever expand fewer total nodes than *breadth-first search* does?

PROBLEM 5 - Decision Trees (40 points)

Assume you are given the following three features with the possible values shown.

```
F1  ∈  {A, B}
F2  ∈  {C, D}
F3  ∈  {E, G, H}
```

Consider the following training examples.

```
F1 = A    F2 = C    F3 = H      category = +
F1 = A    F2 = D    F3 = G      category = +
F1 = A    F2 = C    F3 = G      category = -
F1 = A    F2 = D    F3 = G      category = +
F1 = B    F2 = D    F3 = H      category = -
F1 = B    F2 = D    F3 = H      category = -
F1 = B    F2 = C    F3 = H      category = +
F1 = B    F2 = C    F3 = G      category = -
```

A. What score would the *information gain* formula assign to each of the three features?
(Note: I(x,y)=I(y,x), I(1,0)=0, I(0.5, 0.5)=1, I(0.75, 0.25)=0.81, I(0.875, 0.125)=0.54)
Be sure to show all your work.

B. Which feature would be chosen as the root? __________
(Break any ties in favor of *F1* over *F2* over *F3*, and '+' over '-'.)

C. Show the remaining steps, if any, that the decision-tree learning algorithm would perform using the above examples. Again, be sure to show all your work (use the back of this or the previous sheet if necessary).

D. What kind of *search strategy* is used in the decision-tree learning

algorithm? ________________________

Explain your answer.

E. Consider altering the decision-tree learning algorithm to instead use *beam search* (with a beam width of two), where information remaining (`Remainder`) is the scoring function. Briefly outline how this could be done.

What is the START node in your design?

What is your GOAL-TEST?

(Continuation of Problem 5E.)

After expanding the start node, what would your OPEN list contain when using the above training examples?

Problem 6 - Information Theory and Expected Values (10 points)

Assume you have a training set for a given task that contains *three* different output categories, where three-quarters of the examples are in category *A*, and one-eighth are in each of categories *B* and *C*. What is the expected amount of information needed in order to label a future example for this task? Explain the derivation of any formulae you use. (You needn't provide a numerical answer; it is fine to provide an expression containing some *log*'s in it.)

How do we expect to get this needed information?