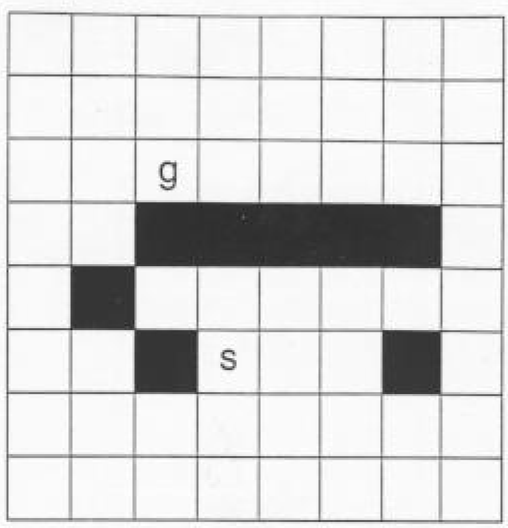
C1 Artificial Intelligence (25 points)
Search

Consider the problem of finding a path from position "s" to position "g" in the grid shown below. A piece can move on the grid horizontally and vertically, one square at a time. No step may be made into a forbidden shaded area.

1) (7 points) Using extra grid paper available from the exam proctor, number the nodes visited (in order) for a depth−first search from s to g, given that the order of operators you will test is : up, left, right, then down. Assume there is a cycle check. A node is visited when you take it off the open list or frontier. What is the first path found?

2) (9 points) Using another grid, number the nodes visited, in order, for greedy search from s to g. The heuristic function should be Manhattan Distance. The Manhattan Distance between two points is the distance in the x−direction plus the distance in the y−direction (i.e. the number of moves given only up/left/right/down moves−−−the Manhattan Distance corresponds to the distance traveled along city streets arranged in a grid). Assume that you automatically prune any multiple paths. Assume that moves with the same value are arranged on the open list/frontier so that when several moves have the same value, the newest moves will be tried first in the up/left/right/down order. What is the first path found?

3) (9 points) Number the nodes in order visited for A*, with multiple path pruning. Make the same assumptions, and use the heuristic as in problem 2. What is the first path found?



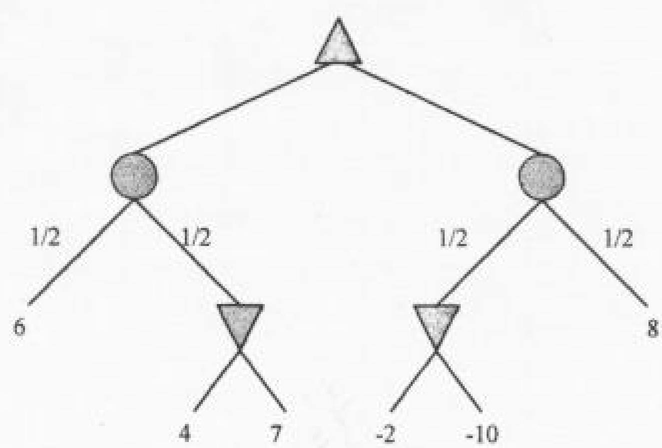
C2 Artificial Intelligence (25 points)
Game Playing

a) (5 points) Draw the smallest possible game tree on which alpha-beta will prune at least one leaf node. Make sure to label the leaves with values and circle the leaf (or leaves) that will be pruned.

b) (9 points) Minimax search assumes a zero-sum game. This means that a single utility function is used by both players, and that a utility of x for MAX means a utility of -x for MIN. Describe how the minimax algorithm should change when we have nonzero-sum games -- that is, when each player has his or her own utility function. You may assume that each player knows the other's utility function.

c) (5 points) Discuss whether or not alpha-beta pruning can be done with the algorithm discussed in b) above. That is, when in a situation where it is a nonzero-sum game and the utility function of MAX is independent of the utility function for MIN, can alpha-beta pruning be done?

d) (6 points) Consider a situation where a game involves an element of chance (e.g., the rolling of dice). The following diagram shows a game tree with chance nodes and evaluations at the leaves. MAX nodes are represented by triangles pointing up, MIN nodes by triangles pointing down, and chance nodes are represented by circles. The fraction on the line below each chance node represents the probability of that outcome occurring. Assuming that a minimax search is done on the tree below (calculating expected values at the chance nodes), copy the diagram and fill in the values of all internal nodes including the root, and indicate the best move at the root.



C3 Artificial Intelligence (25 points)
Planning

The briefcase domain consists of five operators:

open()
close()
take−out(x)
put−in(x)
carry(from,to)

1) (10 points) Describe these five operators in a POP/STRIPS−style planning language. Note: to describe the operator *carry(from,to)* correctly, you should use a logical conditional and universal quantification in the effect description. That is, each and every item that is in the briefcase will move when the briefcase is carried from one location to another.

2) (6 points) These operators, together with a description of an initial state and a goal, constitute a planning problem. Assume that there are two items in the world (besides the briefcase **B**): a paycheck **P** and a dictionary **D**. All 3 are initially located at HOME, and the paycheck is inside the closed briefcase, and the dictionary is not. The goal is to have the paycheck at home and the dictionary at the OFFICE. Describe the START and FINISH operators corresponding to these initial conditions and goals.

Let *bopen, ~bopen* indicate that the briefcase is open/closed
at(x,loc) indicate that *x* is at location *loc*
in(x,B) indicate that *x* is inside the briefcase *B*.

3) (9 points) Draw a complete plan for this problem.

C4 Artificial Intelligence (25 points)
Decision Trees

Consider the criteria for accepting candidates to the Ph.D. program at the mythical University of Get-A-Degree. Each candidate is evaluated according to four attributes: the grade point average (GPA), the quality of the undergraduate university attended, the publication record, and the strength of the recommendation letters. To simplify our example, let us give each attribute discrete values and limit the possible value of each attribute: possible GPA scores are 4.0, 3.7, and 3.5; universities are categorized as top_10, top_20, and top_30; publication record is a binary attribute -- either the applicant has published previously or not; and recommendation letters are similarly binary, they are either good or normal. Finally the candidates are classified into two classes: accepted (or P for 'positive'), and rejected (or N for 'negative').

Pat Applicant does not know the decision tree used by University Get-A-Degree. However, Pat, through a friend in the admissions office, does have the following data regarding twelve of last year's applicants:

No.	Attributes				Class
	GPA	UNIVERSITY	PUBLISHED	RECOMMENDATION	
1	4.0	Top_10	Yes	Good	P
2	4.0	Top_10	No	Good	P
3	4.0	Top_20	No	Normal	P
4	3.7	Top_10	Yes	Good	P
5	3.7	Top_20	No	Good	P
6	3.7	Top_30	Yes	Good	P
7	3.7	Top_30	No	Good	N
8	3.7	Top_10	No	Good	N
9	3.5	Top_20	Yes	Normal	N
10	3.5	Top_10	No	Normal	N
11	3.5	Top_30	Yes	Normal	N
12	3.5	Top_30	No	Good	N

C4 Artificial Intelligence (continued)

a) (13 points) Use a decision tree algorithm (with information gain computations for selecting split variables) to induce the root node of a decision tree for University of Get-A-Degree using the examples in the above table. Show as much of the tree as possible after inducing only the root (i.e., if any branch classifies all of the instances the same, show what decision would be reached; if a branch does not classify all instances the same, simply show the remaining instances under the branch). Show all of the computations involved. Some calculations that might be useful: $I(x,y)=I(y,x)$, $I(1,0)=0$, $I(1/2,1/2)=1$, $I(1/3,2/3)=0.92$, $I(1/4,3/4)=0.81$, $I(1/5,4/5)=0.72$, $I(2/5,3/5)=0.97$, $I(1/6,5/6)=0.65$, $I(1/7,6/7)=0.59$, $I(2/7,5/7)=0.86$, $I(3/7,4/7)=0.98$, $I(1/8,7/8)=0.54$, $I(3/8,5/8)=0.95$.

b) (5 points) Imagine that our training examples contained a discrete-valued feature that had a unique value for each example, such as an applicant's social security number. What would the information gain of such a feature be? Briefly discuss the implications of your answer on inducing a decision tree.

c) (7 points) Briefly discuss the concept of pruning in a decision tree algorithm. What problem is it intended to solve? In general terms describe how pruning is used.