Midterm Preparation.

The midterm will test all material, not only material after the midterm. Notice that the following questions are a sample of the kind of questions that could be on the midterm. However, they certainty do not cover everything that could be on the midterm. They are merely intended to give you some idea of the kind of questions that you can expect. The homeworks also give you an idea of the kind of questions that you can expect. In general, you probably want to look at all of the assignments (including sample solutions), handouts, slides, and the book. Please do not forget to bring a calculator to the midterm.

You want to solve a map coloring problem: given the map of some country, you want to color its states so that neighboring states have different colors, using a total of 5 colors. Which AI techniques can be applied to this problem, and why?

TRUE of FALSE?

No search method that makes use of heuristic functions can GUARANTEE to find a shortest path from the start to a goal: true or false.

Breadth-first search always terminates if there is a path from the start to the goal: true or false.

Depth-first search always terminates if there is a path from the start to the goal: true or false.

A* is an informed version of depth-first search, that is, a version of depth-first search that uses a heuristic function to bias the search towards the goal states: true or false.

Given two admissible heuristic functions, at least one dominates the other one: true or false.

It is possible to use A* with a heuristic function that assigns zero to every state: true or false.

Suppose that you have a search problem where you are more concerned with minimizing the search effort than with minimizing the solution cost. Which of the following f-values would you use in conjunction with the A* method? f(s) = 0, f(s) = g(s), or f(s) = h(s).

Depth-first search and uniform cost search behave identically if all actions have a cost of one, all else being equal: true or false.

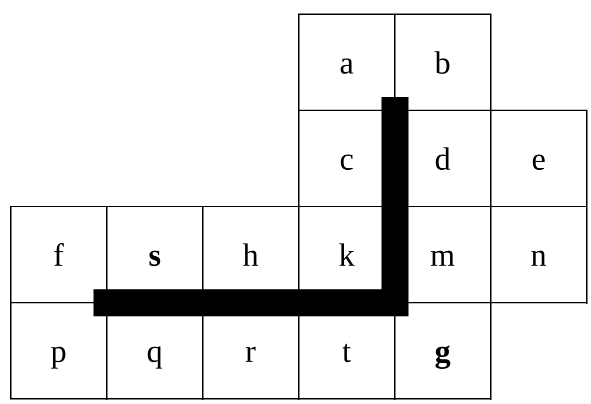
In the following maze the successors of a cell include any cell directly to the east, south, west or north of the current cell except that no transition may pass through the central barrier. For example successors(m) = {d, n, g}. The search problem is to find a path from s to g. We are going to examine the order in which cells are expanded by various search algorithms. For example, one possible expansion order that breadth-first search might use is: s h f k p c q a r b t d g. There are other possible orders depending on which of two equal-distance-from-start states happen to be expanded first. For example s f h p k c q r a t b g is another possible answer.

a) Assume you run depth-first search until it expands the goal node. Assume that you always try to expand East first, then South, then West, then North. Assume your version of depth-first search avoids loops: it never expands a state on the current path. What is the order of state expansion?

b) You decide to use a Manhattan-Distance heuristic function, where h(state) = shortest number of steps from state to g if there were no barriers. So, for example, h(k) = 2, h(s) = 4, and h(g) = 0. Is this heuristic function admissible? Is it consistent?

c) Assume that you use best-first greedy search using heuristic h (a version that never re-explores the same state twice). Again, give all the states expanded, in the order they are expanded, until the algorithm expands the goal state.

d) Finally, assume you use A* search with heuristic h, and run it until it terminates using the conventional A* termination rule. Again, give all the states expanded, in the order they are expanded. (Note that depending on the method that A* uses to break ties, more than one correct answer is possible).



You want to solve a traveling salesperson problem (find the shortest tour for a given set of cities that connects all of them). How could this problem be solved with A*? How could it be solved with hillclimbing? Compare the advantages and disadvantages of the two approaches.

Model a simple world in STRIPS in which there are rooms with doors that can be either open (and unlocked), closed and locked, or closed and unlocked. There is an agent that can go to different rooms, pick up some or all of the keys that are in its current room, take them with it, use them to lock and unlock doors, and drop them anywhere. The agent cannot go through closed doors and thus might have to open and close them, as well as unlock and lock them.

Define the terms “machine learning” and “inductive learning.”

Describe a learning situation that you would solve with neural networks but not genetic algorithms and one learning situation that you would solve with genetic algorithms but not neural networks. Explain your choices in detail and discuss which properties of learning situations make them suitable for neural networks and which properties make them suitable for genetic algorithms.

Describe problems that can result for the decision tree learning method discussed in class when some features have more than two values.

Assume that you have the following training examples available:
(t, t, f) is t
(f, f, f) is t
(t, f, f) is f
Use all of the training examples to construct a decision tree using the decision tree learning method discussed in class. Why is the test at the root node of the decision tree not uniquely determined?

Construct by hand a neural network that computes the XOR function of two inputs. Make sure to specify what sort of units you are using. Why can't a single perceptron compute the XOR function? Can a decision tree represent the XOR function? (If so, show one. If not, explain why not.)

Why isn't it common to use networks of perceptrons that use linear activation functions? Why isn't it common to use networks of perceptrons that use step functions as activation functions?

Explain how crossvalidation can be used in the context of a) decision trees and b) neural networks.

What are the advantages and disadvantages of solving STRIPS-type planning problems with search methods (such as versions of A*)? In this context, what are the advantages and disadvantages of searching forward from the start state rather than backward from the goal state?

Explain in which way backpropagation is hillclimbing (for example, what is the state space, the neighbors of the current state, and the objective function)?