HÁSKÓLINN Í REYKJAVÍK REYKJAVÍK UNIVERSITY	T-622-ARTI, Introduction to Artificial Intelligence Spring 2009 Problem Set 1 Worth 5% of final course grade Due by 23:59 on Monday the 16th of February

Student Name: ____

1. (40%) Formulating Search Problems

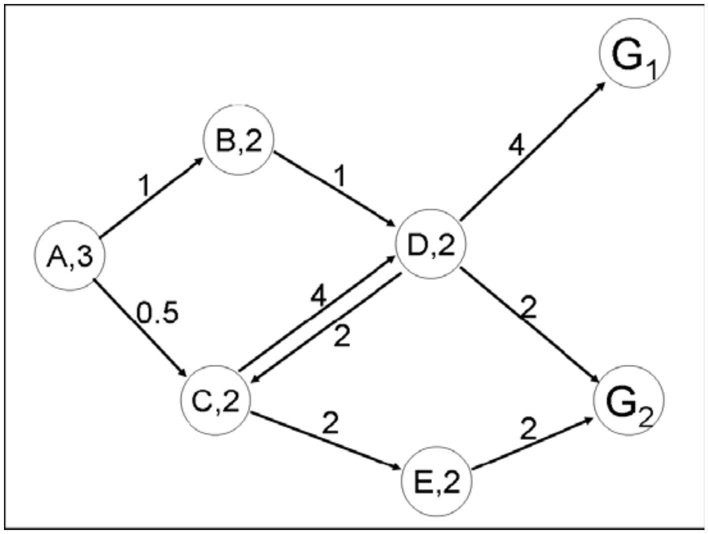
Formulate the following search problems by giving an *initial state*, *goal test*, *successor function*, and *cost function*. Choose a formulation that is precise enough to be implemented at your chosen level of abstraction.

a. An agricultural robot needs to plant 100 trees in a valley. The trees should be at least 1 m apart and should generally stand in a flat area if possible. The robot carries all the trees in a large container on its back.

b. A space probe has to find some place outside of earth for humanity to colonize. The humans it reports to want to be able to choose from at least 3 habitable places.

2. (20%) Performing an A* Search

The following state-space is given, with each state showing its *state id* and the value of its heuristic estimate (*h(s)*) of how far it is from a goal state. Edges indicate the successor function and their actions are labeled with their cost. The start state is *A* and both G1 and G2 are valid goal states.



Use the A* algorithm to find the optimal path to a goal state. Tie-breaking of equal *f*-values on the *Open list (fringe)* is done alphabetically (lower values first). A search node consists of the *state id*, *g-value*, *h-value*, and the *parent state id*, e.g. *[A, 0, 3, -]*. The table below shows the open and closed lists when tracing an execution of the algorithm. Continue the trace until you have identified a path to a goal state. Show the contents of both the open and closed list after each (expansion) step.

Open List (fringe)	**Closed List**
[A, 0, 3, -]	
[B, 1, 2, A], [C, 0.5, 2, A]	[A, 0, 4, -]

3. (20%) Inventing Heuristics

The „RushHour" puzzle involves freeing a car from a traffic jam by moving other cars out of the way. The puzzle is played on a grid where each car can occupy one or more squares along the vertical or along the horizontal axis. A car can move into any unobstructed and free squares in front of it or in back of it. Note that moving a car to a free spot is considered *a single move*, regardless of the distance. Given these restrictions on movement, the question is what is the least amount of shuffling needed to get your car out. A good demonstration of this puzzle is given here: http://janim.net/java-games/rushhour/rushhour.html

Imagine that you need to solve this puzzle using an A* search (no need to actually solve it). What would be a good heuristic function to choose? Provide arguments. Show that your function is *admissable*. Is it also *consistent*?

4. (20%) Formulating a CSP Problem

Formulate the following problem as a CSP problem. Indicate clearly what the *variables* are, the *domains* of the variables, and the *constraints* (no need to solve the problem).

You are trying the new revolutionary CSP (Coffee, Sweets & Pizzas) diet course to get into shape for the summer. Instead of worrying about calorie intake as most diets do, this diet is about making sure you consume enough of the three goods specified above (plus, of course, anything else you want!). The only restrictions are on the coffee, sweet, and pizza consumption: you may consume only one of the three in any given day and not the same in two consecutive days. Additionally, no sweets are allowed on a Saturdays (bad for your teeth), and no coffee on Fridays (you can get too hyper watching your favorite TV show). The diet offers miracle results in only 7 days. The task is to set up a valid diet plan for the week (Mon-Sun).