INSTRUCTIONS

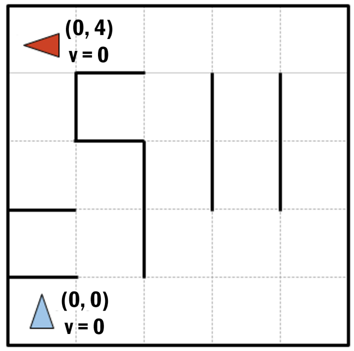
- **Due: Tuesday, 28 January 2020 at 10:00 PM EDT.** Remember that you may use up to 2 slip days for the Written Homework making the last day to submit **Thursday, 30 January 2020 at 10:00 PM EDT**.
- **Format:** Submit the answer sheet pdf containing your answers. You should solve the questions on this handout (either through a pdf annotator, or by printing, then scanning). Make sure that your answers (typed or written neatly) are within the dedicated regions for each question/part. If you do not follow this format, we may deduct points.
- **How to submit:** Submit a pdf with your answers on Gradescope. Log in and click on our class 15-281 and click on the submission titled HW2 and upload your pdf containing your answers.
- **Policy:** See the course website for homework policies and Academic Integrity.

Name	
Andrew ID	
Hours to complete?	

For staff use only

Q1	Q2	Q3	Q4	Total
/30	/15	/35	/20	/100

Q1. [30 pts] Search and Heuristics



Imagine a car-like agent wishes to exit a maze like the one shown above. The agent is directional and at all times faces some direction $d \in (N, S, E, W)$. With a single action, the agent can *either* move forward at an adjustable velocity v or turn. The turning actions are *left* and *right*, which change the agents direction by 90 degrees. Turning is only permitted when the velocity is zero (and leaves it at zero). The moving actions are *faster*, *maintain* and *slower*. *Faster* increments the velocity by 1 and *slower* decrements the velocity by 1; in both cases the agent then moves a number of squares equal to its NEW adjusted velocity. Maintaining velocity allows the agent to go just as many squares as last time. Any action that would result in a collision with a wall crashes the agent and is illegal. Any action that would reduce v below 0 or above a maximum speed V_{max} is also illegal. The agents goal is to find a plan which parks it (stationary) on the exit square using as few actions (time steps) as possible.

(a) [4 pts] Suppose the agent wants to take the leftmost path from the start (0,0) facing north to the goal (0,4) facing west. Write down the sequence of actions for it to take.

Actions:

(b) [4 pts] If the grid is M by N, what is the size of the state space? You should assume that all configurations are reachable from the start state.

State Space Size:

(c) [4 pts] What is the maximum branching factor of this problem? Draw an example state (x, y, orientation, velocity, grid/walls) that has this branching factor, and list the set of available actions.

You may assume that illegal actions are simply not returned by the problem model. You do not necessarily have to use the example grid above. Make sure to label your drawing.

Maximum Branching Factor:

Maximum Branching Example State and Available Actions:

What is the minimum branching factor? Draw an example state (x, y, orientation, velocity, grid/walls) that has this branching factor, and list the set of available actions.

You may assume that illegal actions are simply not returned by the problem model. You do not necessarily have to use the example grid above. Make sure to label your drawing.

Minimum Branching Factor:

Minimum Branching Example State and Available Actions:

(d) [4 pts] Is the Manhattan distance from the agents location to the exits location admissible?

If not, draw an example state (x, y, orientation, velocity, grid/walls) where this heuristic overestimates at that state, and specify: 1) the heuristic value at that state and 2) the actual cost from that state to the goal.

You may assume that illegal actions are simply not returned by the problem model. You do not necessarily have to use the example grid above. Make sure to label your drawing, including the goal state.

◯ Yes ◯ No

Example State, Heuristic Value, Actual Cost:

(e) [4 pts] Is the following heuristic admissible? *The number of turns for the agent to take to face the goal.*

If not, draw an example state (x, y, orientation, velocity, grid/walls) where this heuristic overestimates at that state, and specify: 1) the heuristic value at that state and 2) the actual cost from that state to the goal.

You may assume that illegal actions are simply not returned by the problem model. You do not necessarily have to use the example grid above. Make sure to label your drawing, including the goal state.

◯ Yes ◯ No

Example State, Heuristic Value, Actual Cost:

(f) [4 pts] Is the following heuristic admissible? *Manhattan distance* / V_{max}.

If not, draw an example state (x, y, orientation, velocity, grid/walls) where this heuristic overestimates at that state, and specify: 1) the heuristic value at that state and 2) the actual cost from that state to the goal.

You may assume that illegal actions are simply not returned by the problem model. You do not necessarily have to use the example grid above. Make sure to label your drawing, including the goal state.

◯ Yes ◯ No

Example State, Heuristic Value, Actual Cost:

(g) [2 pts] If we used an inadmissible heuristic in A* search, could it change the completeness of the search?

Answer:

(h) [2 pts] If we used an inadmissible heuristic in A* search, could it change the optimality of the search?

Answer:

(i) [2 pts] Give a general advantage that an inadmissible heuristic might have over an admissible one.

Answer:

Q2. [15 pts] Search Nodes

Consider tree search (i.e. no explored set) on an arbitrary search problem with max branching factor b. Each search node n has a backward (cumulative) cost of $g(n)$, an admissible heuristic of $h(n)$, and a depth of $d(n)$. Let n_c be a minimum-cost goal node, and let n_s be a shallowest goal node.

For each of the following, give an expression that characterizes the set of nodes that are explored before the search terminates. For instance, if we asked for the set of nodes with positive heuristic value, you could say: for all n, such that $h(n) \geq 0$. Don't worry about ties (so you wont need to worry about $>$ versus $\geq$). If there are no nodes for which the expression is true, you must write "none."

(a) [5 pts] Give an inequality in terms of the functions defined above to describe the nodes n that are explored in a breadth-first search before terminating.

Inequality: All n, such that:

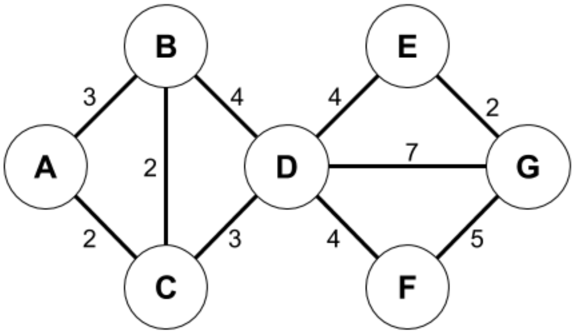
(b) [5 pts] Give an inequality in terms of the functions defined above to describe the nodes n that are explored in a uniform cost search before terminating.

Inequality: All n, such that:

(c) [5 pts] Now for this question, assume the heuristic is consistent. Give an inequality in terms of the functions defined above to describe the nodes n that are explored in an A* before terminating.

Inequality: All n, such that:

Q3. [35 pts] Searching a Graph



Node	h_1	h_2
A	12.5	11
B	12	10
C	11	9
D	5	6
E	1	2
F	4	4.5
G	0	0

Consider the state space graph shown above. A is the start state and G is the goal state. The costs for each edge are shown on the graph. Each edge can be traversed in both directions. Please refer to the search algorithms **exactly as presented on the lecture slides** as the ordering of the actions matters.

(a) [21 pts] For each of the following **graph search** strategies, mark with an X which (if any) of the listed paths it could return. Note that for some search strategies the specific path returned might depend on tie-breaking behavior. In any such cases, make sure to mark **all** paths that could be returned under some tie-breaking scheme.

Algorithm	A-B-D-G	A-C-D-G	A-C-D-E-G
BFS	(i)	(ii)	(iii)
DFS	(iv)	(v)	(vi)
UCS	(vii)	(viii)	(ix)
Greedy with heuristic h_1	(x)	(xi)	(xii)
Greedy with heuristic h_2	(xiii)	(xiv)	(xv)
A* with heuristic h_1	(xvi)	(xvii)	(xviii)
A* with heuristic h_2	(xix)	(xx)	(xxi)

(b) [1 pt] What is the cost of the optimal path for uniform cost search from A to G?

Answer:

(c) [4 pts] Is h_1 admissible? Is it consistent?

Admissible?

Consistent?

(d) [4 pts] Is h_2 admissible? Is it consistent?

Admissible?	**Consistent?**

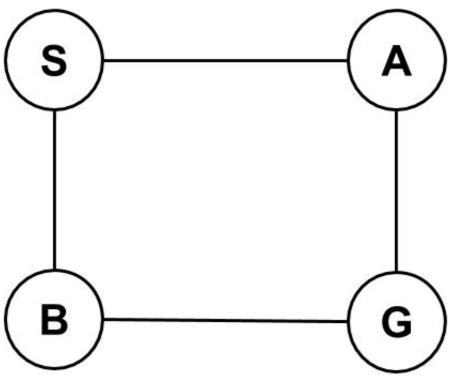
(e) [2 pts] For any given graph, is the path returned by greedy search always more expensive than the path returned by A* search? If you answer yes, explain; If you answer no, provide a simple counterexample. (Assume the heuristic used for A* is consistent and admissible.)

Answer:

(f) [1 pt] When is A* tree search optimal? When is A* graph search optimal?

Tree Search?	**Graph Search?**

(g) [2 pts] Fill in edge cost and heuristic values in the graph below such that uniform cost search and greedy search will return the same path from S to the G. (Hint: You may want to try this on a piece of scratch paper before filling in the values!)



Edge Costs	
Edge	**Cost**
$S-A$	
$S-B$	
$A-G$	
$B-G$	

Heuristic Values	
State	$h(s)$
S	
A	
B	
G	0

Q4. [20 pts] Search: Multiple Choice and Short Answer Questions

(a) [12 pts] Consider the following true/false questions with each question worth 2 points. For the following search problems, assume every action has a cost of at least ϵ, with $\epsilon > 0$. Assume any heuristics used are consistent.

Depth-first tree-search is guaranteed to be complete.
◯ **True** ◯ **False**

Breadth-first tree-search is guaranteed to be complete.
◯ **True** ◯ **False**

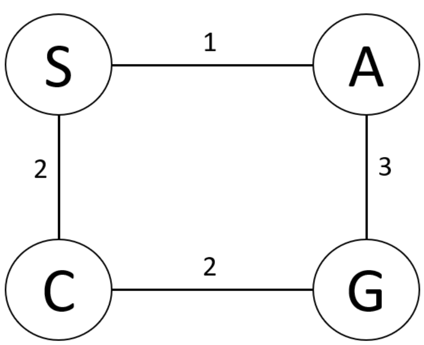
Iterative deepening tree-search is guaranteed to be complete.
◯ **True** ◯ **False**

On graphs without cycles, graph-search contains a larger frontier than tree-search.
◯ **True** ◯ **False**

Iterative deepening graph-search has the time complexity of BFS and the space complexity of DFS.
◯ **True** ◯ **False**

If h_1(s) is a consistent heuristic and h_2(s) is a consistent heuristic, then min(h_1(s), h_2(s)) must be consistent.
◯ **True** ◯ **False**

(b) [8 pts] Consider the state space graph shown below. S is the start state and G is the goal state. The costs for each edge are shown on the graph. For the following table below, fill in potential heuristic values such that the heuristic is admissible but not consistent.



Heuristic Function	
State	$h(s)$
S	
A	
C	
G	0