HW1: Search

CS6300: Artificial Intelligence, Spring 2018 **Jake Pitkin**
University of Utah

1 Graph Search

For the following problems, S will denote Start and G will denote Goal. When choosing an arbitrary order of state expansions, alphabetical ordering will be used. Once visited, a state will not be expanded again.

1. Greedy search

Step	**Frontier**	**Expand**
1	(S, 0)	S
2	(S-A, 3), (S-B, 2)	B
3	(S-A, 3), (S-B-D, 1), (S-B-C, 2), (S-B-G, 4)	D
4	(S-A, 3), (S-B-C, 2), (S-B-G, 4), (S-B-D-G, 5)	C
5	(S-A, 3), (S-B-G, 4), (S-B-D-G, 5), (S-B-C-G, 1)	G

Table 1: Greedy search

Greedy search final path: $S \rightarrow B \rightarrow C \rightarrow G$

2. Depth first search

   Note: as a LIFO queue is used, nodes are enqueued in reverse alphabetical order, so they are expanded in alphabetical order.

Step	**Frontier**	**Expand**
1	S	S
2	A, B	A
3	G, B	G

Table 2: Depth first search

Depth first search final path: $S \rightarrow A \rightarrow G$

3. Breadth first search

   Note: when adding nodes to the frontier, a check is first made to see if they are already contained in the frontier. If they are, they aren't re-added.

Step	Frontier	Expand
1	S	S
2	A, B	A
3	B, G	B
4	G, C, D	G

Table 3: Breadth first search

Breadth first search final path: $S \rightarrow A \rightarrow G$

4. Uniform cost search

Step	Frontier	Expand
1	(S, 0)	S
2	(S-A, 3), (S-B, 2)	B
3	(S-A, 3), (S-B-C, 4), (S-B-D, 3), (S-B-G, 6)	A
4	(S-B-C, 4), (S-B-D, 3), (S-B-G, 6), (S-A-G, 6)	D
5	(S-B-C, 4), (S-B-G, 6), (S-A-G, 6), (S-B-D-G, 8)	C
6	(S-B-G, 6), (S-A-G, 6), (S-B-D-G, 8), (S-B-C-G, 5)	G

Table 4: Uniform cost search

Uniform cost search final path: $S \rightarrow B \rightarrow C \rightarrow G$

5. Greedy search with the heuristic values listed at each state

Step	Frontier	Expand
1	(S, 0)	S
2	(S-A, 2), (S-B, 3)	A
3	(S-B, 3), (S-A-G, 0)	G

Table 5: Greedy search with heuristic values

Greedy search with heuristic final path: $S \rightarrow A \rightarrow G$

6. A* search with the heuristic values listed at each state

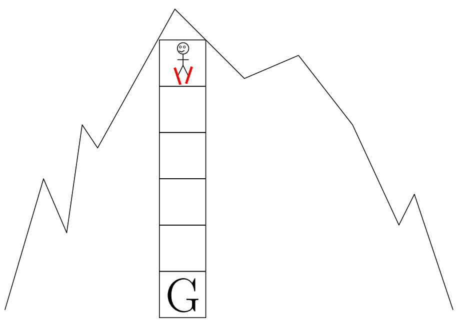
Step	Frontier	Expand
1	(S, 0)	S
2	(S-A, 5), (S-B, 5)	A
3	(S-B, 5), (S-A-G, 6)	B
4	(S-A-G, 6), (S-B-C, 5), (S-B-D, 4), (S-B-G, 6)	D
5	(S-A-G, 6), (S-B-C, 5), (S-B-G, 6), (S-B-D-G, 8)	C
6	(S-A-G, 6), (S-B-G, 6), (S-B-D-G, 8), (S-B-C-G, 5)	G

Table 6: A* search

A* search final path: $S \rightarrow B \rightarrow C \rightarrow G$

2 Downhill Skiing

After getting to Alta, Alice takes the lift up to the top of the mountain. The run is really rocky, so her only option is to go straight downhill. She begins with a velocity of 0 and can safely maintain a maximum velocity of V. At any state, she has three actions she can take: accelerate, decelerate or coast. If the accelerates, her velocity increases by 1; if she decelerates, it decreases by 1; if she coasts, it stays the same. *After* her velocity is adjusted by her action, she moves downhill an equal number of squares to her current velocity.



Consider the above figure. If Alice's first action is "accelerate" then she will end up in the second square down with a velocity of 1. If she then "coasts" then she will end up in the third square down with a velocity of 1. If she "accelerates" again, she will end up in the fifth square down with a velocity of 2.

Alice's goal is to reach the chair lift (marked "G") with a velocity of zero. (No, Alice cannot have negative velocities). She would like to get there as quickly as possible. However, if she has a non-zero velocity at the goal, she skies into the parking lot and destroys her skis.

1. If the mountain is N units tall (eg., it is $N = 6$ units tall in the figure), what is the size of the state space? Justify your answer. (You may ignore "unreachable" states.) What are the start/goal states?

2. Give an example of a state that is not reachable. Suppose that Alice cannot coast (she must either accelerate or decelerate): does this yield *more* unreachable states? If so, give an example of one and justify your answer either way.

3. Is Alice's current elevation (i.e., distance from the chair lift) an admissible heuristic? Why or why not?

4. State and justify a non-trivial, admissible heuristic for this problem which is *not* current elevation.