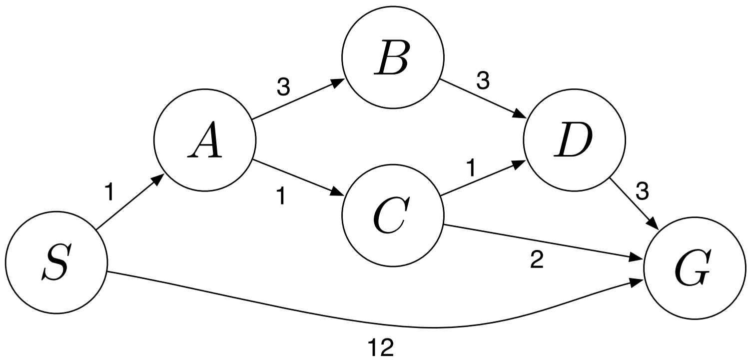
CS 188
Fall 2019

Final Review Search Solutions

Q1. Search



Answer the following questions about the search problem shown above. Assume that ties are broken alphabetically. (For example, a partial plan $S \to X \to A$ would be expanded before $S \to X \to B$; similarly, $S \to A \to Z$ would be expanded before $S \to B \to A$.) For the questions that ask for a path, please give your answers in the form '$S - A - D - G$.'

(a) What path would breadth-first graph search return for this search problem?
$S - G$

(b) What path would uniform cost graph search return for this search problem?
$S - A - C - G$

(c) What path would depth-first graph search return for this search problem?
$S - A - B - D - G$

(d) What path would A* graph search, using a consistent heuristic, return for this search problem?
$S - A - C - G$

(e) Consider the heuristics for this problem shown in the table below.

State	h_1	h_2
S	5	4
A	3	2
B	6	6
C	2	1
D	3	3
G	0	0

(i) Is h_1 admissible? **Yes** $\boxed{\textbf{No}}$

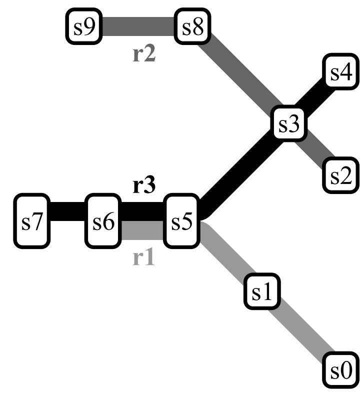
(ii) Is h_1 consistent? **Yes** $\boxed{\textbf{No}}$

(iii) Is h_2 admissible? $\boxed{\textbf{Yes}}$ **No**

(iv) Is h_2 consistent? **Yes** $\boxed{\textbf{No}}$

Q2. Search: Formulation

You are building an app that tells users how to travel between two locations in a subway system as cheaply as possible. You have a map that shows all of the train routes. Two example maps are shown to the right.



Sparse Map
9 Stations
3 Routes

There are R train routes and S stations. It costs \$1 to board a train or to transfer trains. You can transfer between trains if they visit the same station (e.g. in the sparse map you can transfer from route 2 to route 3 at station 3).

To be clear, the cost does not vary for different train routes, it does not depend on where you get on or off each train, or how long you spend waiting at stations, it only depends on the number of trains you catch.

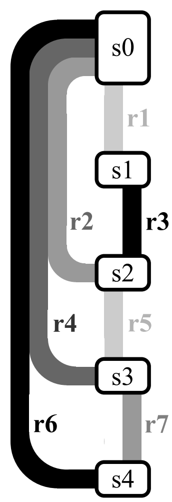
(a) Formulate this problem as a search problem. Choose your definition of states so that the state space is small. Assume R **is larger than** S (a dense map).

(i) States: Each state is represented by a station, the station you are currently at.

(ii) Successor function: Given the current station, it returns stations that can be reached by travelling on a train route from that station. The actions have a cost of \$1.

(iii) Start state: The starting station.

(iv) Goal test: Is the current station the goal station?



Dense Map
5 Stations
7 Routes

(b) Now assume R **is smaller than** S (a sparse map). Formulate this problem as a search problem again, choosing a definition of states so that the state space is small.

(i) States: Each state is represented by a train route, the train route you are currently on.

(ii) Successor function: Given the current train route, it returns all train routes that share a station with that train route. The actions have a cost of \$1.

(iii) Start state: A special train route that visits only the start station.

(iv) Goal test: Does the current train route visit the goal station?