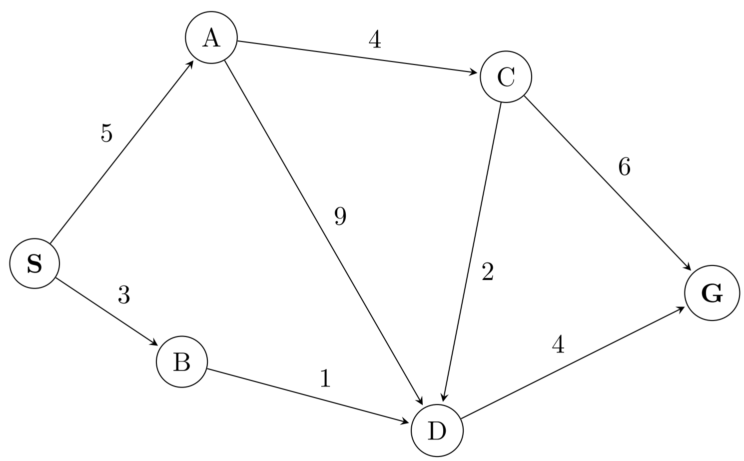
1 Designing & Understanding Heuristics

Today, we will be taking a closer look at how the performance of A^* is affected by the heuristics it uses. To do this, we'll be using the graph below. You may have noticed that no heuristic values have been provided (*Recall:* What is A^* without heuristic values?). This is because we'll be working in pairs to come up with heuristics ourselves!

Please find someone next to you to work with, and decide between yourselves who will design an admissible heuristic and who will design a consistent heuristic. Then, *independently* create these heuristics for the given graph by annotating each node with a heuristic value.

When you have completed your heuristic, exchange your paper with your partner, and work together to answer the questions below.



(a) Write down the path found by the heuristic on the graph above.

(b) Work with your partner to come up with a heuristic that's admissible but not consistent.

(c) Prove that if a heuristic is consistent, it must be admissible.

2 True/False Section

(a) Depth-first search always expands at least as many nodes as A search with an admissible heuristic.

(b) Assume that a rook can move on a chessboard any number of squares in a straight line, vertically or horizontally, but cannot jump over other pieces. Manhattan distance is an admissible heuristic for the problem of moving the rook from square A to square B in the smallest number of moves.

(c) The euclidean distance is an admissible heuristic for Pacman path-planning problems.

(d) The sum of several admissible heuristics is still an admissible heuristic.

(e) Admissibility of a heuristic for A^* search implies consistency as well.