Exercise Sheet: A*, Knapsack problem

Data Structures and Algorithms (X_400614)

1) Perform the A* graph search algorithm on the graph below. The distance between two vertices is given as the weight of the graph. The heuristic $h(n)$ is the value given at the vertex (in red), representing the straight-line distance from that vertex to the goal: vertex G.
   a) Given that you use a *queue*, simulate the queue changes as A* unfolds.
   b) What is the shortest path from vertex S to vertex G?
   c) What is the distance of the shortest path calculated in 1b?

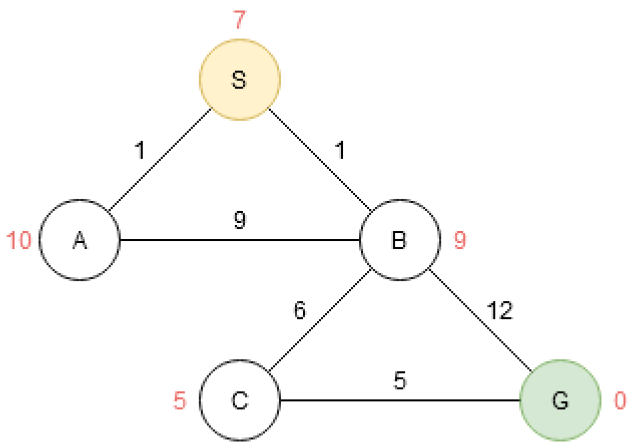
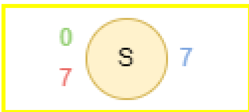
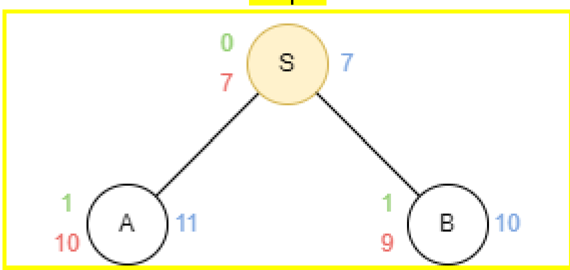


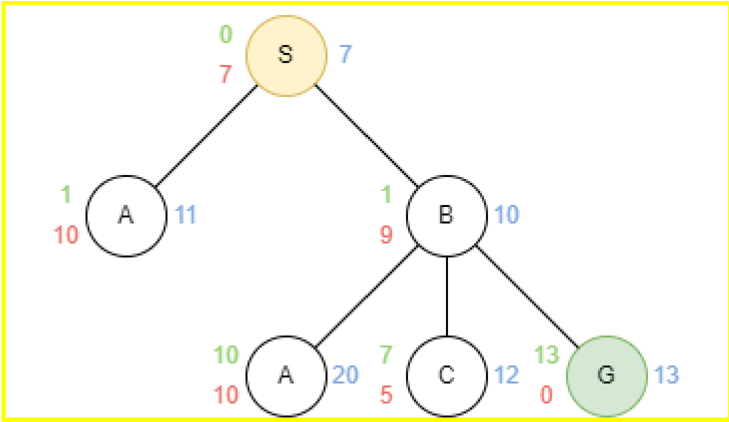
Figure 1. The graph on which to perform A*.



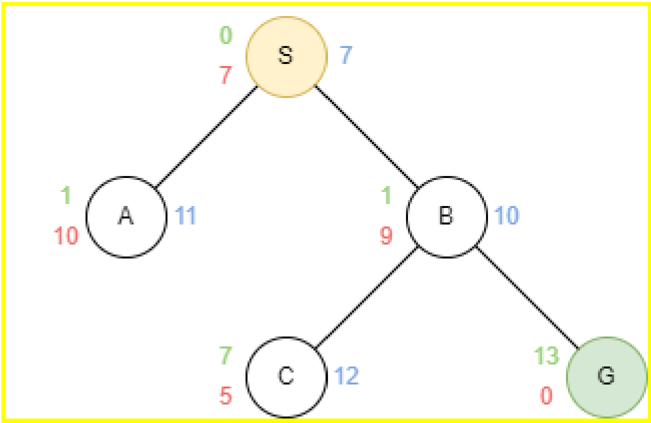
Step 1



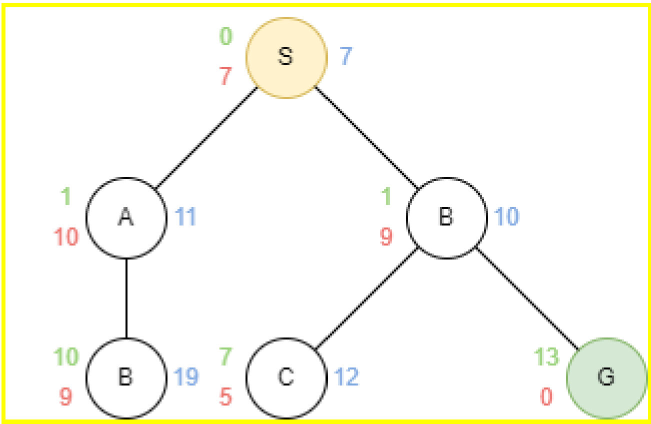
Step 2



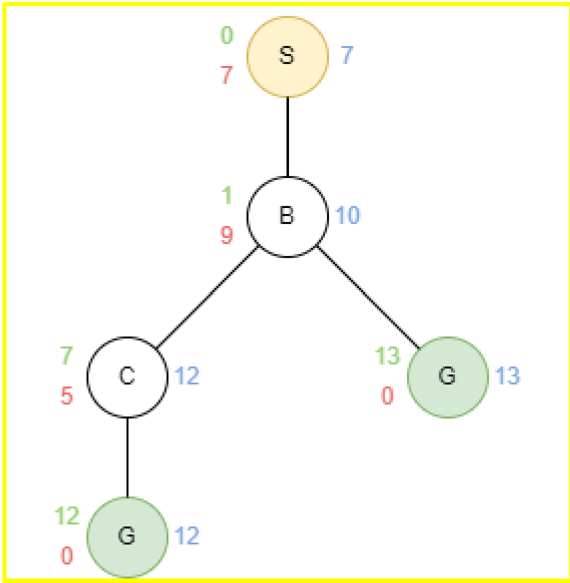
Step 3



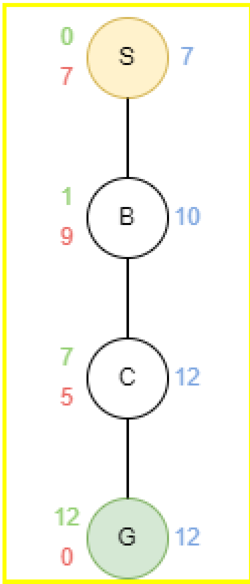
Step 4



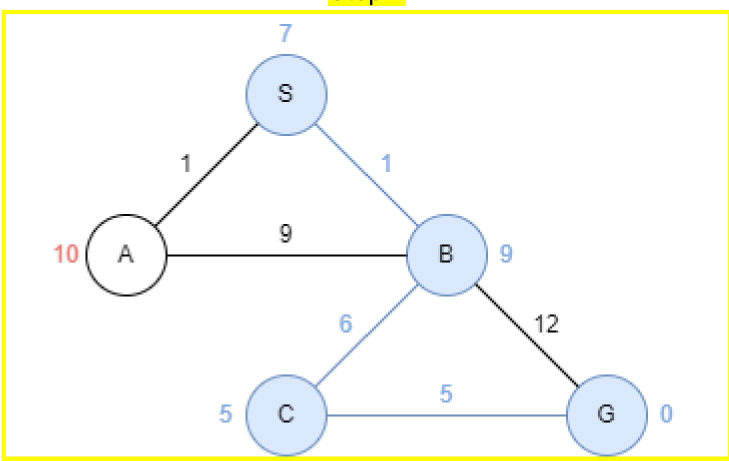
Step 5



Step 6



Step 7



The total distance between S and G is $1 + 6 + 5 = 12$.

2) Answer true or false for the following statements. Assume that the algorithms use a consistent heuristic $h(n)$.

a) A* graph search is guaranteed to return an optimal solution.
True, since the heuristic is consistent in this case.
b) A* graph search is guaranteed to expand no more nodes than depth-first graph search.
False. Depth-first graph search could, for example, go directly to a sub-optimal solution.
c) A* graph search is guaranteed to expand no more nodes than Dijkstra's graph search algorithm.
True. The heuristic could help to guide the search and reduce the number of nodes expanded. In the extreme case where the heuristic function returns zero for every state, A* and Dijkstra's algorithm will expand the same number of nodes. In any case, A* with a consistent heuristic will never expand more nodes than Dijkstra's algorithm.

3) Solve the 0/1 knapsack problem via dynamic programming. The weights for the items are $\{3, 4, 5, 6\}$ and the values for the items are $\{2, 3, 4, 1\}$ and the maximum weight the knapsack can hold is 8.
a) Show the matrix you obtain after applying the algorithm.

	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	0	0	2	2	2	2	2	2
2	0	0	0	2	3	3	3	5	5
3	0	0	0	2	3	4	4	5	6
4	0	0	0	2	3	4	4	5	6

b) Taking which items will maximise your profits?
Taking the items with weights 3 and 5 will maximise your profits of $2 + 4 = 6$.

ACTUAL SOLUTION!!:

3) Solve the 0/1 knapsack problem via dynamic programming. The weights for the items are $\{3, 4, 5, 6\}$ and the profits for the items are $\{2, 3, 4, 1\}$ and the maximum weight of the knapsack is 8.
   a) Show the matrix you obtain after applying the algorithm.
   b) Taking which items will maximise your profits?

Maximum

P	W		0	1	2	3	4	5	6	7	8
–	–	0	0	0	0	0	0	0	0	0	0
2	3	1	0	0	0	2	2	2	2	2	2
3	4	2	0	0	0	2	3	3	3	5	5
4	5	3	0	0	0	2	3	4	4	5	6
1	6	4	0	0	0	2	3	4	4	5	6

5(4) + 3(2) → weight = 8
profit = 6

4) You are given a knapsack that can carry a maximum weight of 60. There are 4 items with weights $\{20, 30, 40, 70\}$ and values $\{70, 80, 90, 200\}$. What is the maximum value of the items you can carry using the knapsack?
a) 160
b) 200
c) 170
d) 90
Answer: a
Explanation: The maximum value you can get is 160. This can be achieved by choosing the items 1 and 3 that have a total weight of 60.