



Princess Sumaya جامعة
University الأميرة سميرة
for Technology للتكنولوجيا

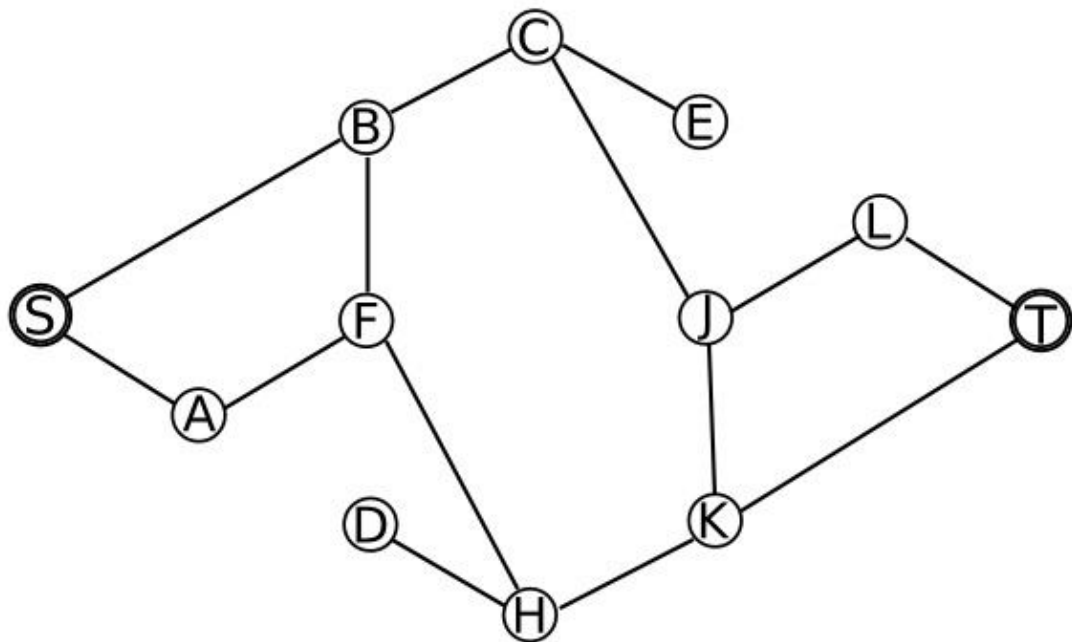
**Princess Sumaya University for Technology
The King Hussein School for Information Technology
Computer Science Department**

Homework 1

Due date is October 27, 2021

Part A (10 pts)

You have just moved to a strange new city, and you are trying to learn your way around. Most importantly, you want to learn how to get from your home at S to the subway at T.



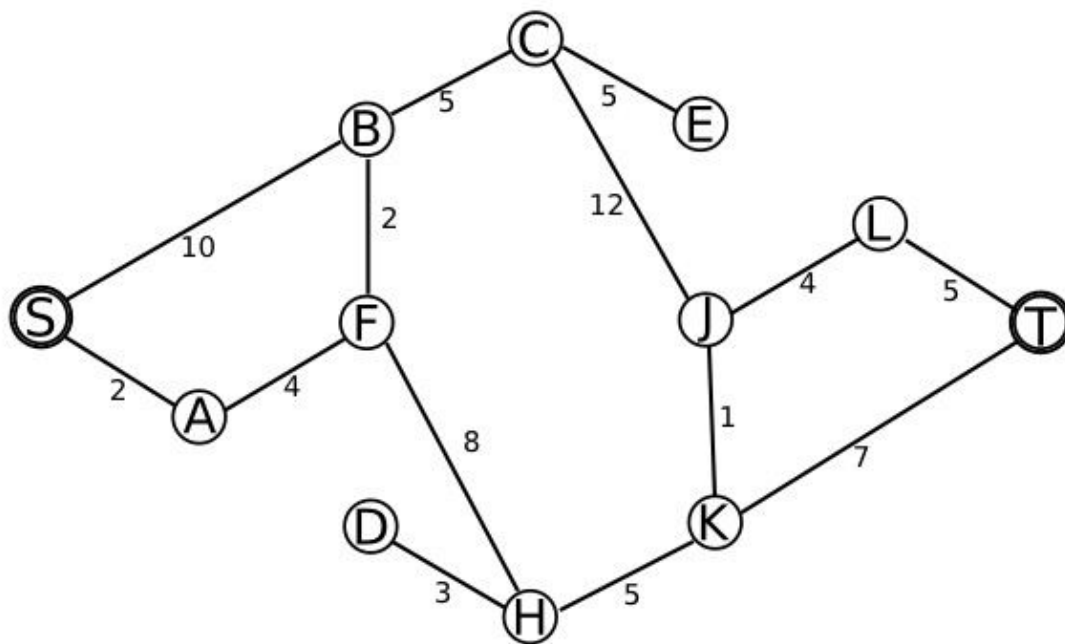
In all search problems, use alphabetical order to break ties when deciding the priority to use for extending nodes.

Using depth-first search with backtracking and **with** an extended list, draw that part of the search tree that is explored by the search.

What is the final path found from the start (S) to the goal (T)?

Part B (15 pts)

Some streets have more traffic on them than others. Your friend who has lived in this city for a long time provides you with information about the traffic on each street - the streets are labeled with costs, in the form of how many minutes it will take you to traverse each one. List the nodes at which you have to backtrack:



Using these given path costs, you are to find the lowest-cost path from S to T using branch-and bound **with an extended list but with no distance heuristic**. First, draw the search tree. **Number each node as it is expanded, from 1 to n.**

Now identify the shortest path:

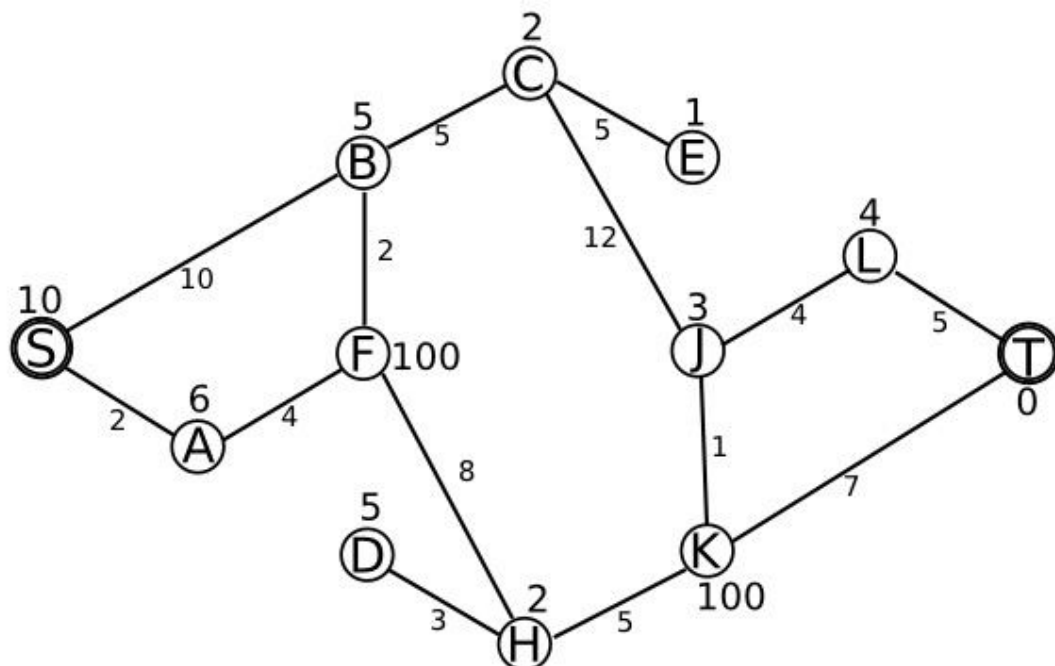
After you have found a path to T, which nodes must you still expand before you can be certain that you have found the shortest path to T?

Part C: (20 points)

Now you are to use A* search, expecting to do less work as you find the lowest-cost path from S to

T. That is, you are to **use both an extended list and a distance**

heuristic. The distance metric is not straight-line distance; instead use the numbers provided by an oracle and written immediately above or below each node. For example, the oracle tells you that the estimated distance from node C to the goal, node T, is 2.



First draw the search tree. **Number each node as it is expanded, from 1 to n.**

Now, show the path you have found:

If the path you found using A* is the same as the path you found in Part B, explain in detail why it must be the same; if the path is not the same, explain why your answers are different.