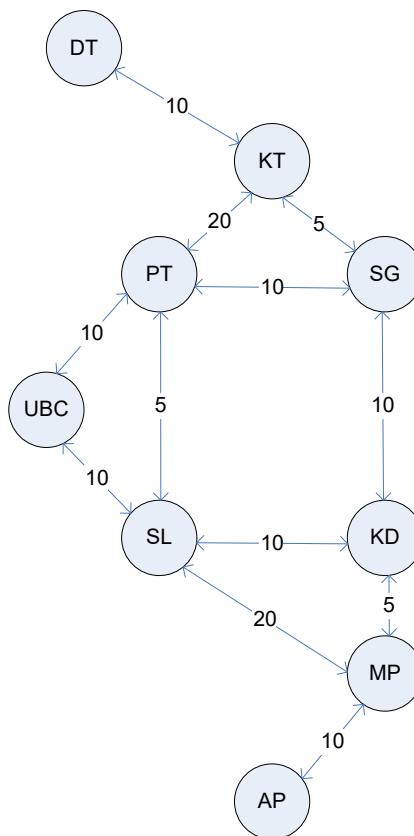


Course Code: CS 401	Course Name: Artificial Intelligence
Instructor Name: Nida Pervaiz	
Student Roll No:	Section:

Question 1

Consider the graph (of driving times in Vancouver; not drawn to scale) with arc lengths shown on the arcs:



Note that the arc costs are for both directions.

For example the arc $_UBC, PT_$ has length 10, and the arc $_PT, UBC_$ has length 10. Suppose the neighbors are given in the following order:

$DT: KT$
 $KT: SG, PT, DT$
 $PT: SL, UBC, KT, SG$
 $SG: KD, PT, KT$
 $UBC: PT, SL$
 $SL: PT, KD, MP, UBC$
 $KD: SG, MP, SL$
 $MP: KD, AP, SL$
 $AP: MP$

We want to find a path from UBC to DT. For each of the following search methods specify whether it halts for this particular graph. If it does halt, specify which solution path is found first and give the path length. If it doesn't halt, briefly explain why.

(a) Depth-first search without loop detection (this means we are not putting a check of visited nodes).

This doesn't halt. It gets stuck in the cycle as its not keeping track of visited and unvisited nodes:

$PT \rightarrow SL \rightarrow PT \rightarrow SL \dots$

(b) Depth-first search with loop detection.

It halts and finds the path:

Search nodes and pathway nodes: $UBC \rightarrow PT \rightarrow SL \rightarrow KD \rightarrow SG \rightarrow KT \rightarrow DT$

The path length is 50.

(c) Breadth-first search.

It halts and finds the path:

Search nodes: UBC -> PT -> SL -> KT -> SG -> KD -> MP -> DT

Pathway: UBC -> PT -> KT -> DT

The path length is 40.

(d) Lowest-cost-first search.

It halts and finds the path:

We will keep track of parent and cost value for each node and based on that pathway found will be

Search nodes (those nodes which will get dequeued from the priority queue until goal node gets dequeued): UBC -> PT -> SL-> SG-> KD-> KT-> MP-> DT

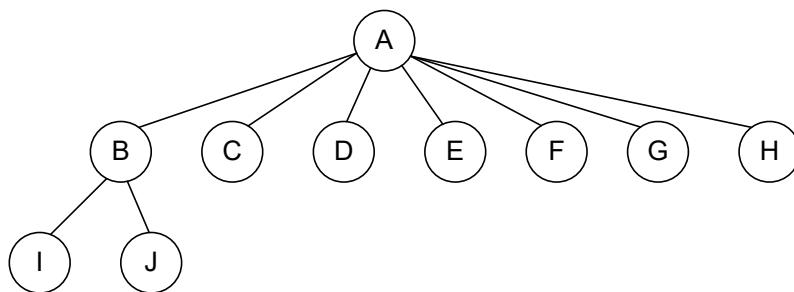
Pathway: UBC -> PT -> SG -> KT -> DT

The path length is 35.

Question 2

Describe a search space (and/or a search tree) in which breadth first search performs much worse than depth first search.

Solution. Make any tree that has more children than levels, like the following:



The goal node is I. In the case of Breadth first, the number of nodes traversed are 8 before the goal is reached, in the case of depth first, the number of nodes traversed is 2.

Question 3

There is a robot placed at the centre of a square grid of 5×5 . The success node is present at bottom extreme right point of the grid i.e. index (5,5). Kindly provide the following details.

Solution.

Initial state (3,3)

Goal state (5,5)

Actions: left, right, up and down

Path cost: 1 step cost as no cost has been mentioned within the question thereby each step takes cost of 1. Hence pathway cost is 4.

State space for one level/iteration:

