Family Name: ....
Other Names: ....

Signature: ....
ID Number: ....

COMP 261 Test

29 April 2015

Instructions

- Time allowed: **45 minutes** .
- Answer **all** the questions. There are 45 marks in total.
- Write your answers in the boxes in this test paper and hand in all sheets.
- If you think some question is unclear, ask for clarification.
- This test contributes 20% of your final grade
- You may use paper translation dictionaries, and non-programmable calculators.
- You may write notes and working on this paper, but make sure your answers are clear.

Questions	**Marks**	
1. Graphs	[20]	
2. Minimum Spanning Trees	[15]	
3. Graphics	[10]	
	TOTAL:	

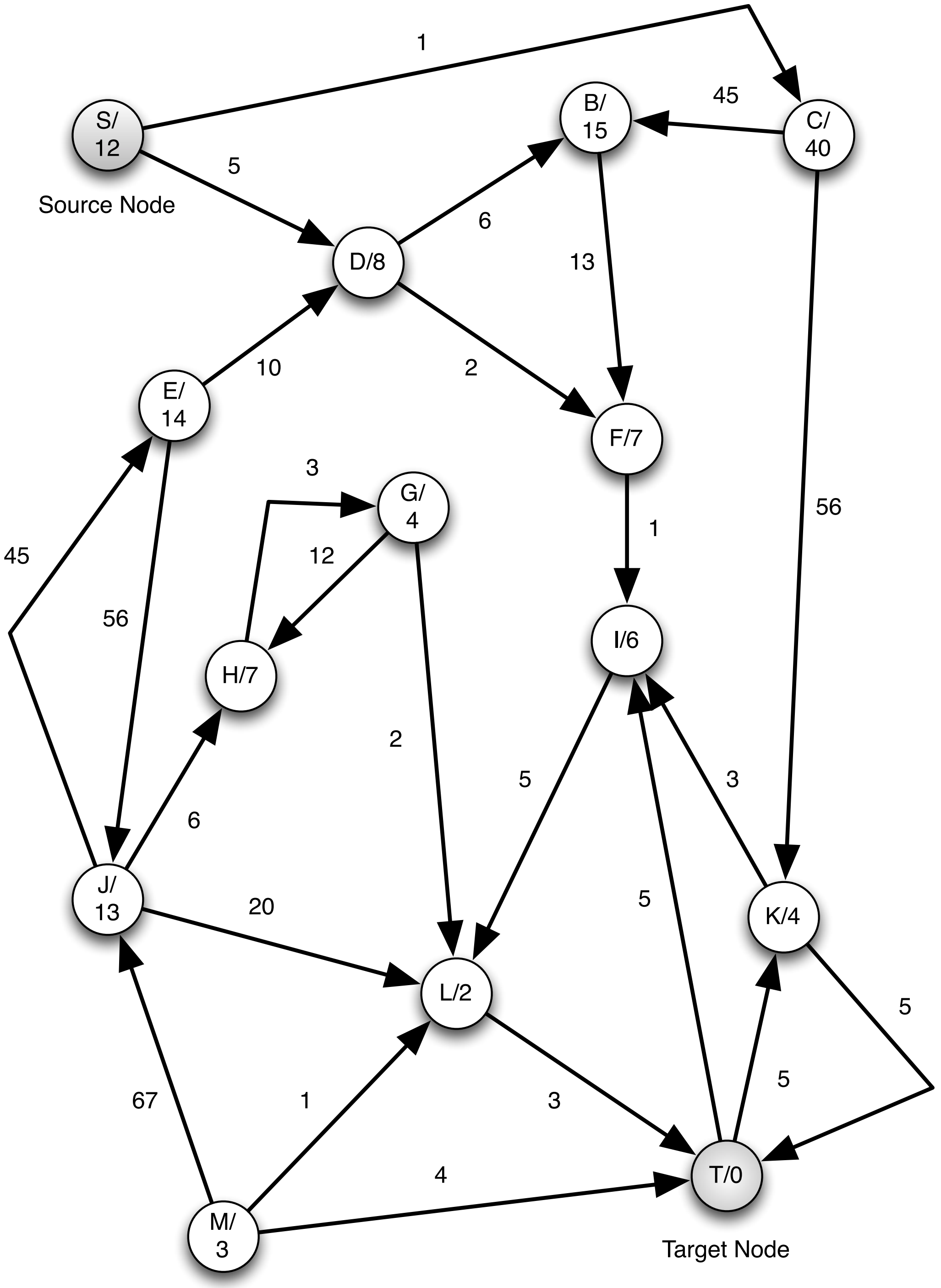
Question 1. Graphs [20 marks]

(a) [5 marks] Fill in the following adjacency matrix (you will need to provide labels) for the directed graph on the facing page (you will also find this graph on the last page of this test that can be torn off for your convenience). Assume that ″edge from node A to node Z″ means a value from ″row A to column Z″.

	S	B	C	D	E	F	G	H	I	J	K	L	M	T
S														
B														
C														
D														
E														
F														
G														
H														
I														
J														
K														
L														
M														
T														

(Question 1 continued on next page)

(Question 1 continued)



(Question 1 continued on next page)

(Question 1 continued)

(b) [5 marks] Given the same graph as in the part (a) show how Dijkstra's single source shortest path algorithm adapted to searching for the shortest path to a given target node (i.e. it stops when it reaches the target) finds the shortest path from S to T.

Below, you should show the queue and the solution:

- list each element that is added to the fringe, showing the node, from-node, and the priority value;
- add neighbours in alphabetical order;
- remove nodes with the same priority value in alphabetical order;
- indicate the order the elements are removed from the fringe (e.g. by numbering them);
- list the nodes of the shortest path found.

Node Name:
From Node:
Priority Value:
Order Removed:

Final Path:

(Question 1 continued on next page)

(Question 1 continued)

(c) [5 marks] Given the same graph as in the parts (a) and (b) show how A* Search finds the shortest path from S to T. The heuristic estimates are provided for you as values inside each node and you can assume they are admissible and consistent.

Below, you should show the queue and the solution:

- list each element that is added to the fringe, showing the node, from-node, and the priority value;
- add neighbours in alphabetical order;
- remove nodes with the same priority value in alphabetical order;
- indicate the order the elements are removed from the frindge (e.g. by numbering them);
- list the nodes of the shortest path found.

Node Name:
From Node:
Priority Value:
Order Removed:

Final Path:

(Question 1 continued on next page)

(Question 1 continued)

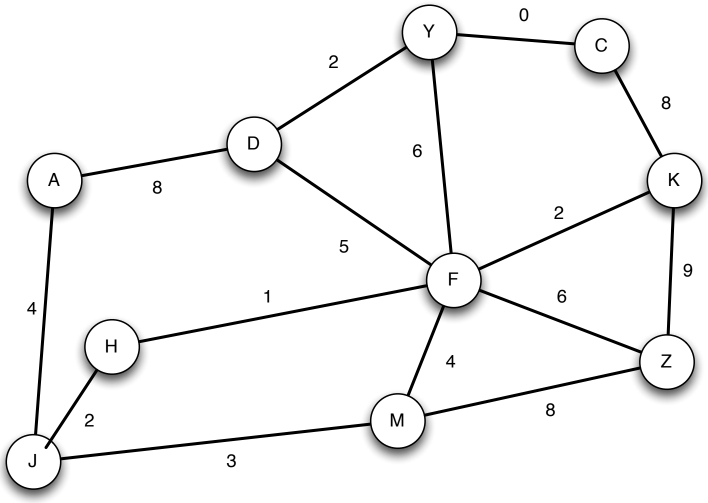
(d) [2 marks] How many steps (i.e. removals from fringe) did your Dijkstra algorithm and A* search took respectively to find the shortest path? If the number of steps is different, explain why it is so and which ones is better and why and if it is the same - again explain why it is so.

(e) [3 marks] Give an example of a small graph, including a start node and a goal node, and an *inconsistent* heuristic estimate of remaining path length for each of the nodes and show how A* would find the wrong path to the goal node using this heuristic. Show the path that A* would find and show the shortest path that it should have found.

Note: show all the edge lengths and show the heuristic estimate on each node.

Question 2. Minimum Spanning Trees [15 marks]

(a) [5 marks] Given the following graph show the steps Prim's algorithm would take to find a MST. In particular, list the edges (by stating the two nodes that they connect and the edge weight: e.g. "DE 5" for the edge between D and E with weight 5) in order that they are added to the MST. To make the answer unique and easy to mark, list the nodes in each edge in *alphabetical* order (DE not ED) and add the edges with the same weight by starting with the edge that comes first alphabetically (the word would appear first in the dictionary). Start at node C / edge "CY 0".



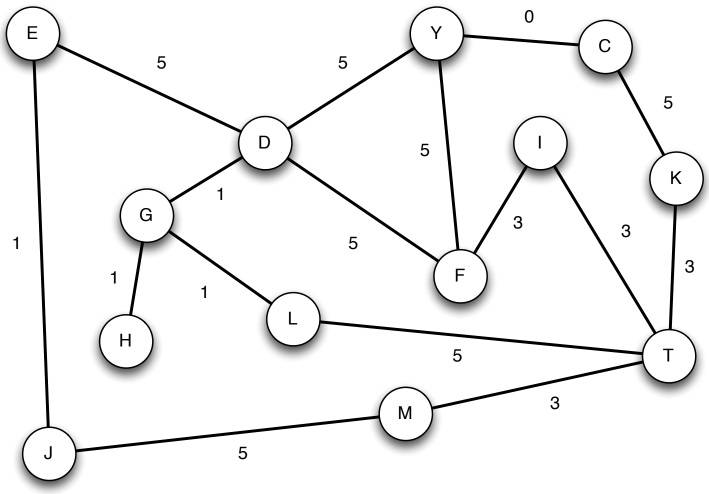
Alphabet:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

(Question 2 continued on next page)

(Question 2 continued)

(b) [5 marks] Given the following graph show the steps Kruskal's algorithm would take to find a MST. In particular, list the edges (by stating the two nodes that they connect and the edge weight: e.g. "DE 5" for the edge between D and E with weight 5) in order that they are added to the MST. To make the answer unique and easy to mark, list the nodes in each edge in *alphabetical* order (DE not ED) and add the edges with the same weight by starting with the edge that comes first alphabetically (the word would appear first in the dictionary). Start at node C / edge "CY 0".



Alphabet:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

(Question 2 continued on next page)

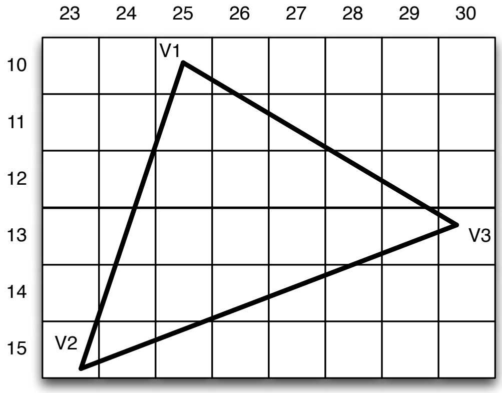
(Question 2 continued)

(c) [5 marks] The above two are known as greedy algorithms. What makes an algorithm greedy? State another greedy algorithm you learned in COMP 261. Describe what it is and explain what makes it greedy.

Question 3. Graphics [10 marks]

(a) [5 marks] Show the output of the Edge List algorithm for the following picture. Show your working (i.e. intermediate or delta values) to ensure you will get all the credit. Round each coordinate to 1 decimal place as appropriate. The three vertices have the following coordinates:

- $V1 = (25, 10, 10)$
- $V2 = (23, 15, 20)$
- $V3 = (30, 13, 30)$



LEFT		RIGHT	
X	Z	X	Z

Your working:

(Question 3 continued on next page)

(Question 3 continued)

(b) [2 marks] Show how to convert to/from 3D and 4D homogeneous coordinates.

(c) [3 marks] State at least two advantages of 4D homogeneous coordinates over standard 3D representation for polygons.

* * * * * * * * * * * * * * *

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

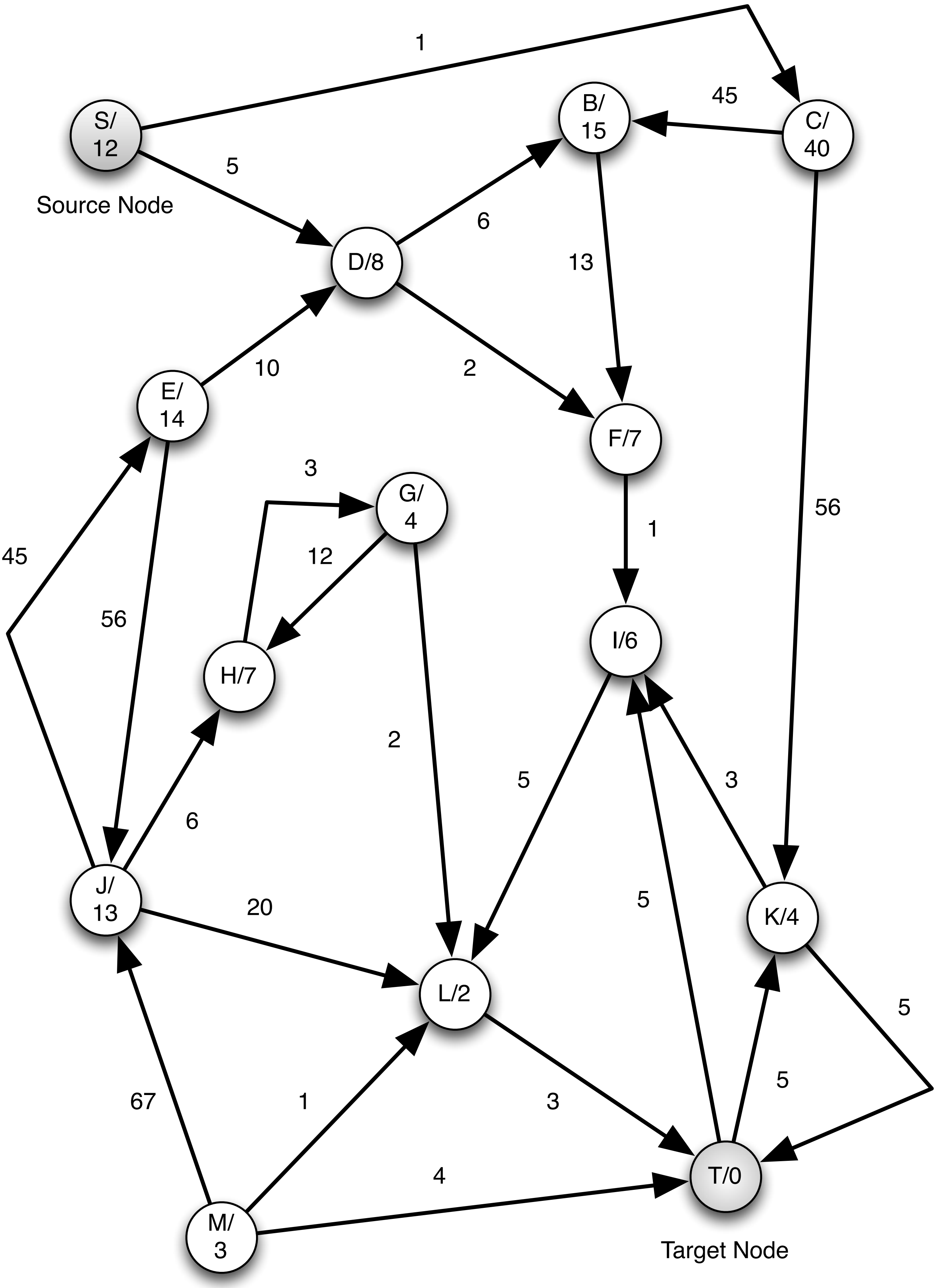


Figure 1: Graph for Question 1