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

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

COMP 261 Test

29 April 2019

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. 3D Graphics	[10]	
	TOTAL:	

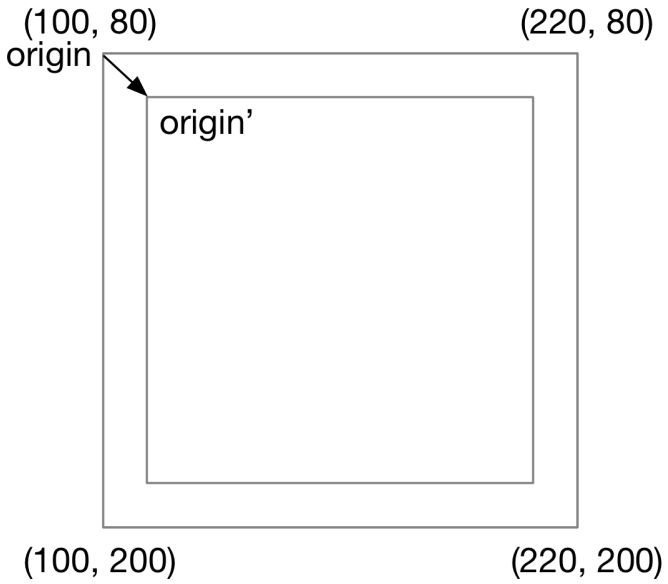
Question 1. Graphs [20 marks]

(a) [6 marks] Answer the following true/false questions about adjacency matrix and adjacency list.

- **Adjacency matrix** must have the same number of rows and columns.
- **Adjacency matrix** of all graphs are symmetric.
- The complexity of checking if an edge exists between two particular vertices in an **adjacency matrix** depends on the number of edges.
- If a graph is stored as an **adjacency matrix**, the time complexity to calculate the number of edges in a graph is $O(E)$, where E is the number of edges in the graph.
- If enumerating all the neighbours of a vertex is the most common operation to the graph, then **adjacency list** is better than **adjacency matrix**.
- If the graph is very sparse, then **adjacency list** is better than **adjacency matrix**.

(b) [4 marks] Below is a display area. The screen is 800x800 pixels. The (x, y) coordinates of the four vertices are (100, 80), (220, 80), (100, 200) and (220, 200). The scaling factor is 1.2. That is, after a zoom in, the scale becomes (1.2 $\times$ scale).

Calculate the (x,y) coordinates of the new origin (the top-left vertex) after a zoom in. Show your working.



(c) [5 marks] We want to store a dictionary containing N number of words. The longest possible word has L characters.

(i) If we use **array list**, what is the worst-case time complexity to add a new word into the dictionary (no check for duplicates)?

(ii) If we use **array list**, what is the worst-case time complexity to search for a word?

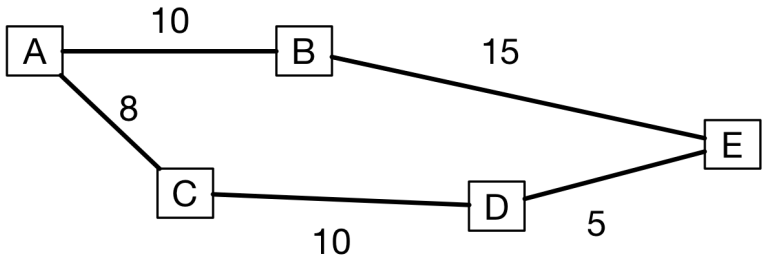
(iii) If we use **trie**, what is the worst-case time complexity to add a new word into the dictionary (no check for duplicates)?

(iv) If we use **trie**, what is the worst-case time complexity to search for a word?

(d) [5 marks] Given the graph below, the nodes are labeled as A, B, C, D, E. Each edge is undirected, and is associated with a number indicating the edge cost.

Show how to use *Dijkstra's algorithm with early stop* to search for the shortest path from node A to node E. You should show (1) at each step, the *elements in the fringe* and the *element to be visited next*, and (2) the *final shortest path* as a sequence of nodes.

Each element is represented in the format of $\langle node, fromNode, costSoFar \rangle$. The information in step 0 is already given for you to start.



Step 0: Fringe elements: $\{\langle A, null, 0 \rangle\}$
Element to visit next: $\langle A, null, 0 \rangle$

Final Path:

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.

Question 2. Minimum Spanning Trees [15 marks]

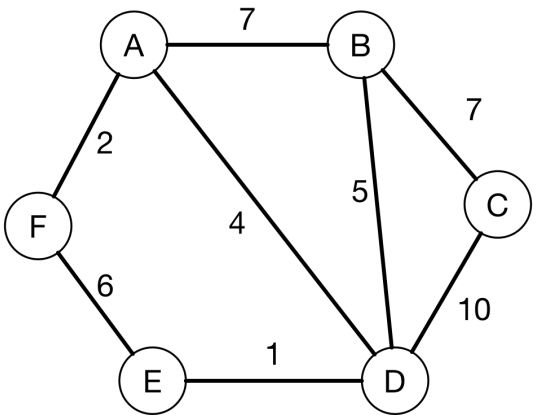
(a) [5 marks] Answer the following true/false questions about minimum spanning tree and disjoint set.

- Every graph has only one minimum spanning tree.
- If a graph has n nodes, then its minimum spanning tree has $n - 1$ edges.
- The correctness of Prim's algorithm depends on the first node it starts from.
- Kruskal's algorithm always finds a minimum spanning tree of a graph.
- Kruskal's algorithm can be efficiently implemented by using the disjoint set data structure.

(b) [5 marks] Given the following graph, show the steps Prim's algorithm would take to find a Minimum Spanning Tree (MST). The algorithm starts at node A. At each step, the algorithm adds one edge into the MST. List the edges in the order that they are added to the MST.

Each edge is represented as the two nodes that they connect and the edge weight: e.g. "EF 4" for the edge between E and F with weight 4.

To make the answer unique and easy to mark, list the nodes in each edge in *alphabetical* order (e.g. DE not ED).



Alphabet:

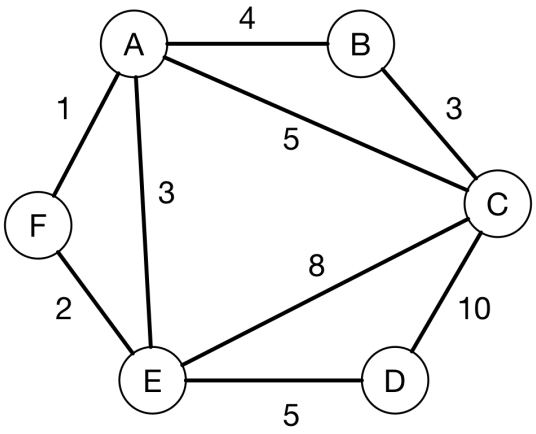
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

(Question 2 continued on next page)

(Question 2 continued)

(c) [5 marks] Given the following graph, show the steps Kruskal's algorithm would take to find a MST. List the edges in the order that they are added to the MST.

Again, list the nodes in each edge in *alphabetical* order (e.g. DE not ED).



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

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.

Question 3. Graphics [10 marks]

(a) [5 marks] Answer the following true/false questions about 3D rendering.

- Triangle is the only shape for polygons in 3D rendering.
- When calculating the coordinates after composite transformation, the order should be "rotation $\rightarrow$ translation $\rightarrow$ scaling".
- If the **normal** of a polygon is pointing towards the viewer, then the viewer can see the polygon.
- Under the coordinate system discussed in the lectures, if the normal of a polygon has a positive z value, then the polygon is visible to the viewer.
- To render a 3D object, it is required to calculate the (x, y, z) coordinates of each pixel covered by each visible polygon of the object.

(b) [2 marks] To render a polygon line by line, we need to calculate the left and right boundaries of each line. This can be done by doing linear interpolation on the three edges (v_1, v_2), (v_2, v_3) and (v_3, v_1) of the polygon. In which of the following cases we are calculating the **left boundary** or $x_{\min}(y)$? *Select all that apply.*

(i) If the vertices are in **anti-clockwise** order, and the edge is **going up**.

(ii) If the vertices are in **anti-clockwise** order, and the edge is **going down**.

(iii) If the vertices are in **clockwise** order, and the edge is **going up**.

(iv) If the vertices are in **clockwise** order, and the edge is **going down**.

(c) [3 marks] The 3D rendering process has the following steps, which are now unordered. Order the steps.

(i) Identify the visible polygons.

(ii) Calculate the normal of each polygon.

(iii) Calculate the coordinates of the vertices after the transformation, if there are any.

(iv) Render by the z-buffer algorithm.

(v) Calculate the shading of the polygons.

Step 1:

Step 2:

Step 3:

Step 4:

Step 5:

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

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.