3.6 Optimisation algorithms

Name: _____

Class: _____

Date: _____

Time: **16 minutes**

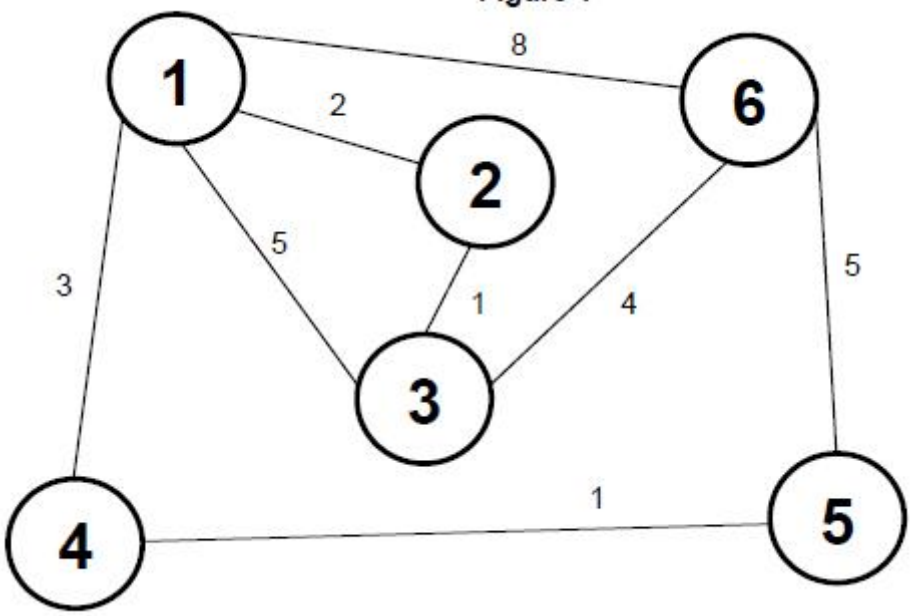
Marks: **16 marks**

Comments:

Q1.

Figure 1 is a graph that shows the time it takes to travel between six locations in a warehouse. The six locations have been labelled with the numbers 1 - 6. When there is no edge between two nodes in the graph this means that it is not possible to travel directly between those two locations. When there is an edge between two nodes in the graph the edge is labelled with the time (in minutes) it takes to travel between the two locations represented by the nodes.

Figure 1



(a) The graph is represented using an adjacency matrix, with the value 0 being used to indicate that there is no edge between two nodes in the graph.

A value should be written in every cell.

Complete the unshaded cells in **Table 1** so that it shows the adjacency matrix for **Figure 1**.

Table 1

	1	2	3	4	5	6
1						
2						
3						
4						
5						
6						

(2)

(b) Instead of using an adjacency matrix, an adjacency list could be used to represent the graph. Explain the circumstances in which it would be more appropriate to use

an adjacency list instead of an adjacency matrix.

_____

_____

_____

_____

(2)

(c) State **one** reason why the graph shown in **Figure 1** is **not** a tree.

_____

_____

(1)

(d) The graph in **Figure 1** is a weighted graph. Explain what is meant by a **weighted graph**.

_____

_____

(1)

Figure 2 contains pseudo-code for a version of Djikstra's algorithm used with the graph in **Figure 1**.

`Q` is a priority queue which stores nodes from the graph, maintained in an order based on the values in array `D`. The reordering of `Q` is performed automatically when a value in `D` is changed.

`AM` is the name given to the adjacency matrix for the graph represented in **Figure 1**.

Figure 2

```
Q ← empty queue

FOR C1 ← 1 TO 6

  D[C1] ← 20

  P[C1] ← -1
  ADD C1 TO Q
ENDFOR

D[1] ← 0
WHILE Q NOT EMPTY

  U ← get next node from Q
  remove U from Q
  FOR EACH V IN Q WHERE AM[U, V] > 0

    A ← D[U] + AM[U, V]
    IF A < D[V] THEN

      D[V] ← A

      P[V] ← U
```

```
    ENDIF
  ENDFOR
ENDWHILE
OUTPUT D[6]
```

(e) Complete the unshaded cells of **Table 2** to show the result of tracing the algorithm shown in **Figure 2**. Some of the trace, including the maintenance of `Q`, has already been completed for you.

Table 2

				D						P					
U	Q	V	A	1	2	3	4	5	6	1	2	3	4	5	6
-	1,2,3,4,5,6	-	-	20	20	20	20	20	20	-1	-1	-1	-1	-1	-1
				0											
1	2,3,4,5,6	2													
		3													
		4													
		6													
2	3,4,5,6	3													
3	4,5,6	6													
4	5,6	5													
5	6	6													
6	-														

(7)

(f) What does the output from the algorithm in **Figure 2** represent?

_____

_____

(1)

(g) The contents of the array `P` were changed by the algorithm. What is the purpose of the array `P`?

_____

_____

_____

_____

(2)

(Total 16 marks)