4.3.6 Optimisation Algorithms

1) **Figure 1** is a graph that shows the time it takes to travel between six cities labelled with the numbers 1 - 6.

When there is an edge between two nodes in the graph the edge is labelled with the time (in hours) it takes to travel between the two cities.

Figure 1

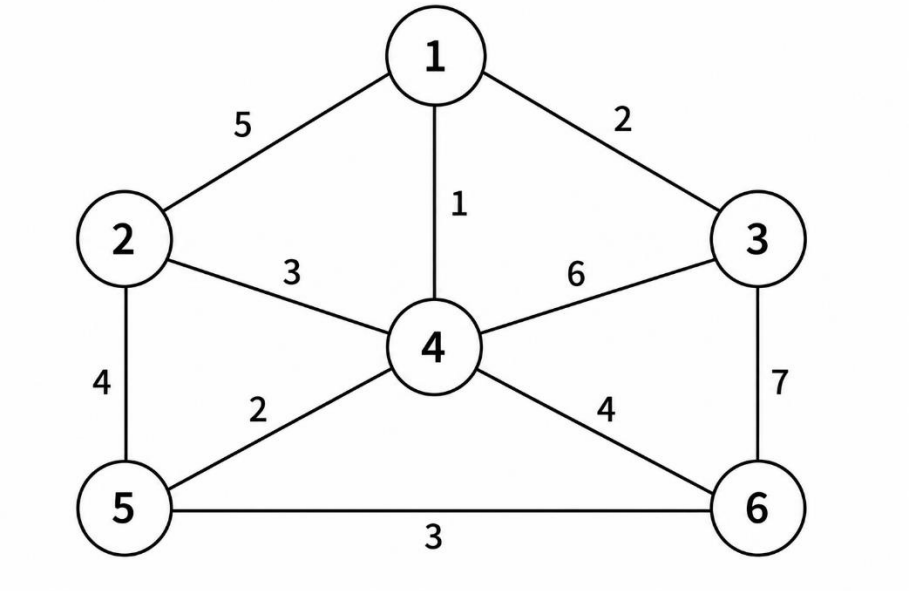

Figure 2 contains pseudo-code for a version of Dijkstra's algorithm used with the graph in **Figure 1**.

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 C ← 1 TO 6
   D[C] ← 20
   P[C] ← -1
   ADD C 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
ENDFOR
OUTPUT D[6]
```

a) Complete the cells of **Table 1** 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 1

				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													
2	3,4,5,6	4													
		5													
3	4,5,6	4													
		6													
4	5,6	5													
		6													
5	6	6													
6	-														

[7 marks]

b) What does the output from the algorithm in Figure 2 represent?

[1 mark]

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

[2 marks]

d) For each of the statements in **Table 2**, complete each row to indicate if the statement is true or false for Dijkstra's algorithm.

Table 2

	True or False?
Calculates the shortest path between a node and other nodes in a graph.	
Can be used to prove that the Halting Problem cannot be solved.	
Can be used with both directed and undirected graphs.	
Can be used with both weighted and unweighted graphs.	

[2 marks]