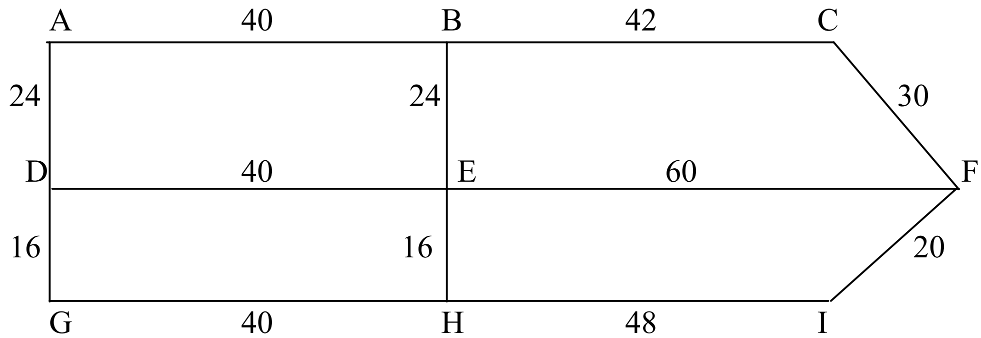
1. It is required to sort the following list of numbers into descending order:

   58 76 55 45 56 34 79

   (i) Use the Bubble Sort algorithm to accomplish this task, giving the state of the list after each pass. [3]

   (ii) An alternative algorithm is to scan the list to find the largest number and put it at the front of the list, then to repeat with the remaining numbers, and so on. Explain why this is generally less efficient than the Bubble Sort method. [3]

2. A prison warder has to patrol along every corridor with cells. This diagram shows a map of one floor of the prison, in which each corridor has cells. The distances are in metres.



   (i) Find the minimum distance that the warder must travel, clearly indicating which corridors he must walk along twice. [5]

   (ii) Write down a possible route of minimum length, starting and finishing at A. [1]

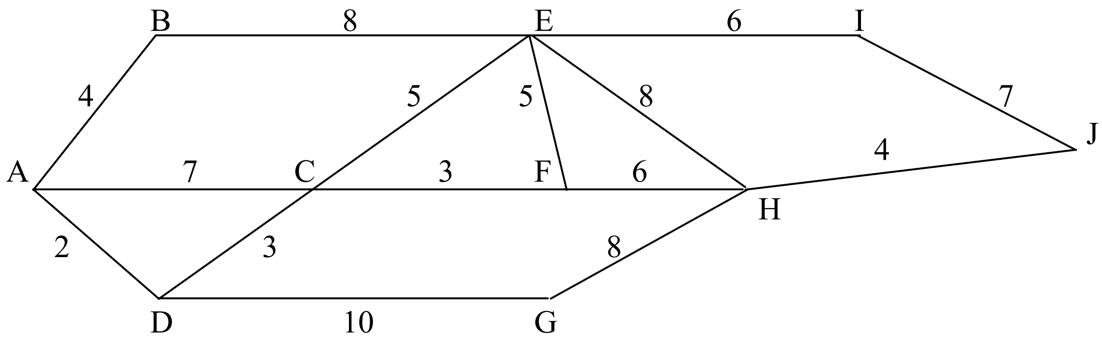
3. The function $P = 2x + 3y$ needs to be maximised, subject to the constraints $2x + y \leq 6$, $x + 4y \leq 8$, $x \geq 0$, $y \geq 0$.

   (i) Set up the initial Simplex Tableau for this problem . [2]

   (ii) Perform one iteration of the Simplex Algorithm, increasing y. [3]

   (iii) Explain the meaning of your answer, in terms of the value of P and the co-ordinates of the final point reached. [2]

4. On a treasure hunt, a competitor wishes to go from A to J as quickly as possible, whilst checking in on the way at E. The times, in minutes, taken to travel between each pair of nodes are as shown.



   (i) Find the quickest route which includes E. [4]

   (ii) Any competitor who fails to check in at E receives 3 penalty points. If each penalty point is equivalent to 1 minute added to the competitor's time, find whether the competitor should bother to visit E at all. [4]

5. One way of solving the Travelling Salesman Problem is to list every possible Hamiltonian cycle.

(i) Show that there are $\frac{(n-1)!}{2}$ such cycles in a complete graph K_n with n nodes. [2]

(ii) If a computer takes 0.03 seconds to enumerate all the cycles for K_{10}, estimate how long it would take for K_{20}. [2]

An alternative method is the Nearest Neighbour Algorithm.

(iii) Show that there are $(n - 1) + (n - 2) + \ldots + 1$ arcs to consider when using this algorithm, and estimate the increase in time required when analysing K_{20} rather than K_{10}. [5]

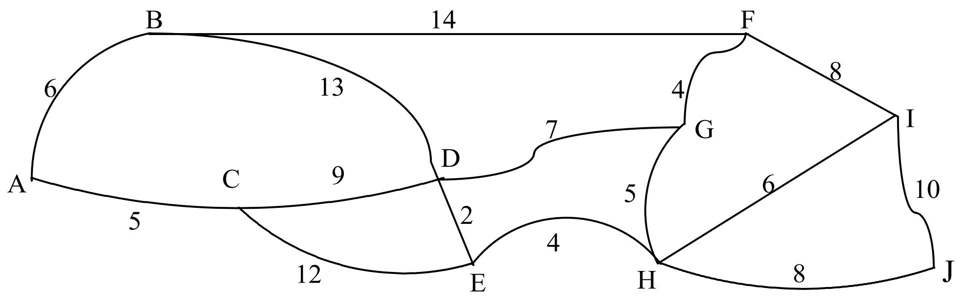
6. A factory makes TV sets and video recorders. The TVs each occupy floor space of 4 square feet, whilst the videos each take 2 square feet. The warehouse has a floor capacity of 1000 square feet. The TVs sell for a profit of £60 and the videos for £45. A TV takes 5 hours to manufacture, with three people working on it, whilst a video takes two people 9 hours. Each video needs two graphics cards, whilst a TV needs just one, and there are 860 cards available each week.
If there are 40 hours in a working week, and 200 people working,

(i) write down three inequalities for x and y, the number of TVs and videos respectively. (Assume that all items must be stored in the warehouse.) [3]

(ii) Draw a graph to illustrate the inequalities, and identify the feasible region. [4]

(iii) Find the maximum profit that may be made, and give the integer values of x and y that generate this profit. [5]

7. State briefly (i) Prim's algorithm, [2]
(ii) Kruskal's algorithm. [2]



For the network shown, find a minimum spanning tree

(iii) using Prim's algorithm, starting at A, and [3]

(iv) using Kruskal's algorithm. [3]

In each case, state the order in which arcs are added.

(v) Give one reason why Prim's algorithm might generally be preferred. [1]

(vi) Give an example of a practical use of a minimum spanning tree. [1]

DECISION MATHS 1 (C) PAPER 4 : ANSWERS AND MARK SCHEME

1. (i)

76	58	55	56	45	79	34		
76	58	56	55	79	45	34	B1	
76	58	56	79	55	45	34		
76	58	79	56	55	45	34	B1	
76	79	58	56	55	45	34		
79	76	58	56	55	45	34	B1	

(ii) The Bubble Sort stops when the numbers are all in the correct order. B1
The alternative algorithm will continue checking each list for the maximum, until the final single number is reached; this may be unnecessary B2 6

2. (i) Odd nodes are B, D, F and H. To eliminate them, make them into even nodes, by connecting in pairs. M1
Possible pairs are : BD + FH = 64 + 68 = 132 A1
BF + DH = 72 + 56 = 128 BH + DF = 40 + 100 = 140 A1 A1
so repeat BF and DH, giving total length of 400 + 128 = 528 m M1
(ii) e.g. A B C F C B E F I H E D E H G D A B1 6

3. (i)

P	x	y	r	s	
1	-2	-3	0	0	0
0	2	1	1	0	6
0	1	4	0	1	8

B2

(ii)

P	x	y	r	s	
1	-1 ¼	0	0	¾	6
0	1 ¾	0	1	-¼	4
0	¼	1	0	¼	2

M1 A1 A1

(iii) When $x = 0$ and $y = 2$, $P = 6$; the algorithm has gone from the (0, 0) vertex of the feasible region to the (0, 2) vertex, but this is still not the optimum value of P. B1 B1 7

4. (i) e.g. do Dijkstra from A to E and E to J M1
For A to E, best solution is ADCE, taking 10 mins; for E to J, best is EHJ, taking 12 mins, giving total of 22 mins. M1 A1 A1
(ii) Direct solution for A to J is ADCFHJ, taking 18 mins; M1 A1
a 3 min. penalty takes this to 21 mins, so it is best to go straight to J M1 A1 8

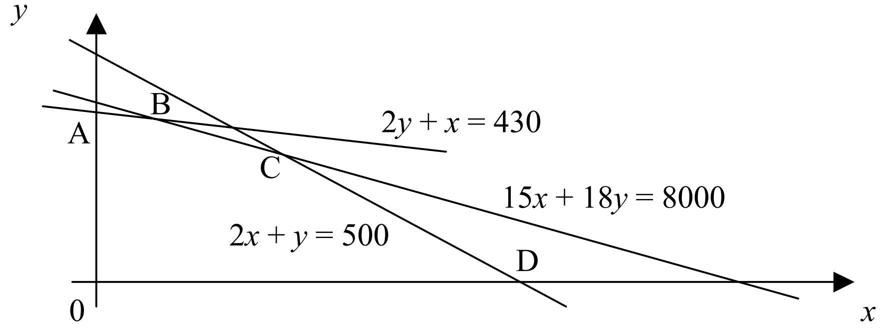
5. (i) From node 1, there are $n - 1$ possible arcs; from each of the nodes then reached, there are $n - 2$ possible arcs etc. This gives a total of $(n - 1)!$ M1
This counts each cycle twice, forwards and backwards, so divide by 2 A1
(ii) Time $\approx (19! / 9!) \times 0.03$ s $\approx 10^{10}$ s ≈ 319 years M1 A1
(iii) Again, $n - 1$ arcs from 1st node; but only choose shortest, then consider B1
$n - 2$ from the 2nd node etc. Then $n - 1 + n - 2 + \ldots + 1 = n(n - 1)/2$ M1 A1
i.e. $O(n^2)$, and going from 10 nodes to 20 nodes increases time by 2^2 M1
i.e. by a factor of 4, to 1.12 s A1 9

6. (i) $4x + 2y \le 1000$ i.e. $2x + y \le 500$ (warehouse space) B1

$15x + 18y \le 8000$ (man-hours in a week) B1

$x + 2y \le 860$ (graphics cards) B1

(ii)



B1 B1 B1 B1

(iii) To maximise $P = 60x + 45y$ Vertices of feasible region, with profits M1

A (0, 430) : 19350 B (43.3, 408.3) : 20975 A1

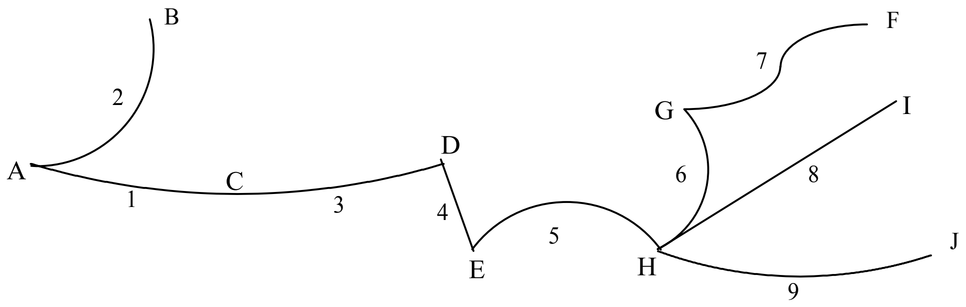
C (47.6, 404.8) : 21071 D (250, 0) : 15000 A1

Integer points near C give (48, 404) as the best solution; $P = 21\,060$ M1 A1 12

7. (i) Prim's algorithm – take any node, find the one nearest to it, and connect. Now find nearest unattached node to those already included, and attach. Repeat until all nodes are attached B2

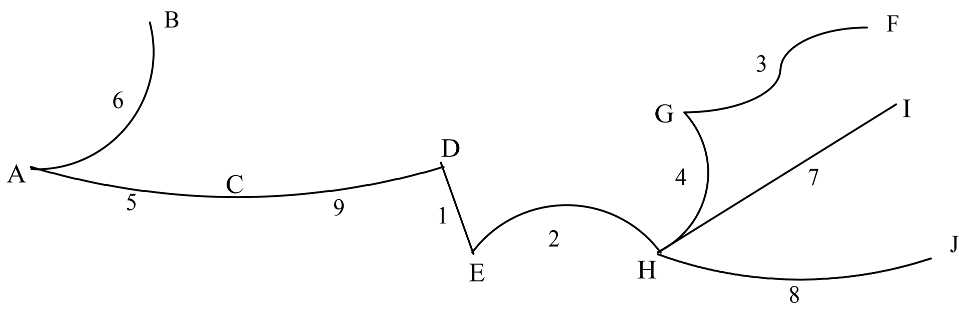
(ii) Kruskal’s algorithm – take shortest arc; take next shortest arc; continue until all nodes are linked, checking each time to ensure that the extra arc does not complete a cycle B2

(iii)



M1 A1 A1

(iv)



M1 A1 A1

(v) Kruskal’s algorithm requires checking, at each stage, to ensure that no cycles have been formed; this checking can be a very lengthy process in a large network B1

(vi) Example, e.g. connecting a network of computers together B1 12