<table>
<tr><th>Q</th><th>Scheme</th><th>Marks</th><th>AOs</th><th>Pearson Progression Step and Progress Descriptor</th></tr>
<tr><td rowspan="2">1a</td><td>A spanning tree such that the total length of its arcs (or edges) are as small as possible.</td><td>B1</td><td>1.2</td><td>3rd<br>Understand minimum spanning trees</td></tr>
<tr><td></td><td>(1)</td><td></td><td></td></tr>
<tr><td rowspan="2">1b</td><td>Any 2 of
<table>
<tr><td>Prim’s</td><td></td><td>Kruskal’s</td></tr>
<tr><td>Graph always connected</td><td>or</td><td>Graph does not have to be connected</td></tr>
<tr><td>Starts with a node/vertex</td><td>or</td><td>Starts with an arc/edge (of least weight)</td></tr>
<tr><td>Can be used with a matrix</td><td></td><td></td></tr>
<tr><td>No need to check for a cycle</td><td></td><td>Must check for cycles</td></tr>
</table></td><td>B2</td><td>1.2</td><td>3rd<br>Understand minimum spanning trees</td></tr>
<tr><td></td><td>(2)</td><td></td><td></td></tr>
<tr><td colspan="5">(3 marks)</td></tr>
<tr><td colspan="5">Notes</td></tr>
</table>

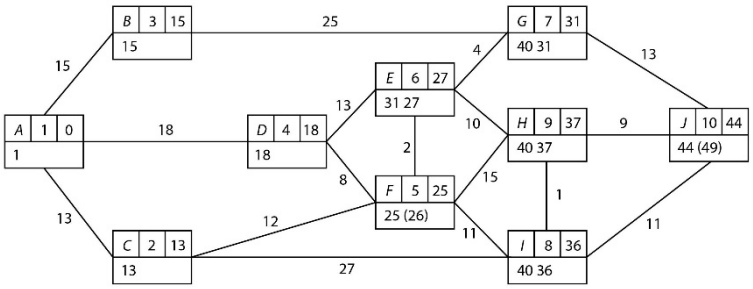
<table>
<tr><th>Q</th><th>Scheme</th><th>Marks</th><th>AOs</th><th>Pearson Progression Step and Progress Descriptor</th></tr>
<tr><td rowspan="2">2a</td><td>
<table>
<tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>G</td><td>H</td></tr>
<tr><td>A</td><td>-</td><td>19</td><td>20</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>B</td><td>19</td><td>-</td><td>-</td><td>11</td><td>13</td><td>18</td><td>-</td><td>-</td></tr>
<tr><td>C</td><td>20</td><td>-</td><td>-</td><td>13</td><td>-</td><td>-</td><td>27</td><td>-</td></tr>
<tr><td>D</td><td>-</td><td>11</td><td>13</td><td>-</td><td>7</td><td>-</td><td>23</td><td>-</td></tr>
<tr><td>E</td><td>-</td><td>13</td><td>-</td><td>7</td><td>-</td><td>17</td><td>-</td><td>15</td></tr>
<tr><td>F</td><td>-</td><td>18</td><td>-</td><td>-</td><td>17</td><td>-</td><td>-</td><td>10</td></tr>
<tr><td>G</td><td>-</td><td>-</td><td>27</td><td>23</td><td>-</td><td>-</td><td>-</td><td>22</td></tr>
<tr><td>H</td><td>-</td><td>-</td><td>-</td><td>-</td><td>15</td><td>10</td><td>22</td><td>-</td></tr>
</table>
</td><td>B2</td><td>1.1b</td><td>5th<br>Apply Prim's algorithm to a distance matrix</td></tr>
<tr><td></td><td>(2)</td><td></td><td></td></tr>
<tr><td rowspan="2">2b</td><td>AB, BD, DE, CD, EH, FH, GH<br><br>Weight 97</td><td>M1<br>A2<br>B1</td><td>1.1b</td><td>5th<br>Apply Prim's algorithm to a distance matrix</td></tr>
<tr><td></td><td>(4)</td><td></td><td></td></tr>
<tr><td colspan="5">(6 marks)</td></tr>
<tr><td colspan="5">Notes<br>2a Allow one error for B1<br>2b M1 for first four arcs or first five nodes correctly chosen in order. A1 for first six arcs or all eight nodes correctly chosen in order. A1 for all seven arcs correctly chosen in order.</td></tr>
</table>

| Q | Scheme | Marks | AOs | Pearson Progression Step and Progress Descriptor |
|---|---|---|---|---|
| **3ai** | *GH*,*AB*,*BE*,*CE*, reject *BC*, {*CD* & *EF* either order}, reject *AC*, *CG*, *HI*, reject *GI*, *IJ* | **M1**<br>**A2** | 2.4 | 4th<br>Apply Kruskal's algorithm to find a minimum spanning tree |
| | | **(3)** | | |
| **3aii** | *AB*, *BE*, *CE*, {*CD* & *EF* either order}, *CG*, *GH*, *HI*, *IJ* | **M1**<br>**A2** | 1.1b | 4th<br>Apply Kruskal's algorithm to find a minimum spanning tree |
| | | **(3)** | | |
| **3b** | Kruskal's<br>Start off the tree with *AD* and *AB* then continue with algorithm. | **B1**<br>**B1** | 2.2a<br>2.4 | 7th<br>Solve problems involving minimum spanning trees in unfamiliar contexts |
| | | **(2)** | | |
| | | | | **(8 marks)** |

**Notes**

**3ai** **M1** for first six arcs correct and in correct order, and evidence of at least one rejection. **A1** for all nine arcs chosen correctly and in correct order. **A1** for all selections and rejections correct, in correct order and at correct time.

**3aii** **M1** for first five arcs correct, in correct order and no rejections, or first six nodes correct in correct order. **A1** for first seven arcs correct or all eight nodes correct and in correct order. **A1** for all arcs correct and in correct order.

| Q | Scheme | Marks | AOs | Pearson Progression Step and Progress Descriptor |
|---|---|---|---|---|
| **4a** |  <br>*ACFEGJ*<br>44 km | **M1**<br>**A1**<br>**A1**<br>**A1**<br>**A1**<br>**A1** | 1.1b | 4th<br>Apply Dijkstra's algorithm to find the shortest path between two vertices |
| | | **(6)** | | |
| **4b** | *ACFEGJ*<br>For example, 44-13=31*JG*, 31-4=27 *GE*, 27-2=25 *EF*, 25—12=13 FC, 13-13=0 *CA*<br>**OR** generic Trace back from *J* including arc *XY* if (*Y* already lies on the path and) the difference of the final values of *X* and *Y* equals the weight of arc *XY* | **B2** | 2.4 | 4th<br>Find the route corresponding to the shortest path |
| | | **(2)** | | |
| **4c** | *ADFEGJ*<br>45 km | **M1**<br>**A2** | 3.2a | 5th<br>Understand how to modify the shortest path problem in different contexts |
| | | **(3)** | | |
| **4d** | *ACFIHJ*<br>46 km | **M1**<br>**A1** | 3.2a | 5th<br>Understand how to modify the shortest path problem in different contexts |
| | | **(2)** | | |
| | | | | **(13 marks)** |

**Notes**

**4a** **M1** Order of labelling boxes filled in with numbers 1-10 and final value boxes filled; **A1** (*ACBDF*) **A1**(*EGI*) **A1**(*HJ*)

**4b** **B1** incomplete explanation but some values shown.

**4c** **A1** correct route, **A1** Correct length, **M1** Incorrect route avoiding *CF* with appropriate answer