Solve the traveling salesman problem with distances in the following table.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M 15</td>
<td>22</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>M 11</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>17</td>
<td>M 20</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>14</td>
<td>8</td>
<td>7</td>
<td>M</td>
</tr>
</tbody>
</table>

Branch-and-bound tree:
(Constructed as we go through the algorithm)

Node 1
Bound 49
1→3

Node 2
Bound 45
4→3

Node 3
Bound 51
3→1

Node 4
Bound 48
1→2
2→4

Node 5
Bound 48

(c) Ensure every column contains at least one zero. Subtract smallest number in each column from all numbers in the column, and add that number to L.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M 7</td>
<td>15</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>M 4</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>9</td>
<td>M 6</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>0</td>
<td>0 M</td>
<td></td>
</tr>
</tbody>
</table>

Previous value of L: 38
Numbers we subtracted from columns: 9 + 8 + 7 + 14 = 38

(d) Ensure every row contains at least one zero. Rows 2 and 4 already contain zeroes; don't need to change them. Subtract 4 from row 1 and 3 from row 3.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M 3</td>
<td>15</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>M 4</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>6</td>
<td>M 3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>0</td>
<td>0 M</td>
<td></td>
</tr>
</tbody>
</table>

Previous value of L: 38
Numbers we subtracted from rows: 4

This is the opportunity matrix for Node 0.
Its L-value is 45.

1. Compute opportunity matrix.
   (a) Set L = 0.
   (b) Ensure every row and column has exactly one M: 

2. [First special case does not apply; opportunity matrix is not 2x2.] 
3. [Second special case does not apply; there is no zero in the opportunity matrix that is the only number in its row or the only number in its column.]

(Continued)
(Processing Node 0)

4. Compute regrets:

5. Choose largest regret: Largest.
   (Corresponds to the link 4→3.)

6. Create child nodes, Nodes 1 and 2.
   (a) Right child (Node 2): label is 4→3.
      Bound is 45 (L-value of opp. matrix)
      Incoming matrix for Node 2:

   [Delete row 4 and column 3.]
   1 2 4
   M 3 0
   O M 0
   3 0 6 3
   \[ L = 45. \]

   (b) Left child (Node 1): label is 4→3.
      Bound is 45+3 = 48.
      (L-value of opp. matrix plus largest regret)
      Incoming matrix for Node 1:

   1 2 4
   M 3 0
   O M 0
   3 0 6 3
   \[ L = 45. \]

5. Choose largest regret. There is a tie
   between 1→2 and 3→1. Go back
   to original distance matrix and compare
   those distances: 1→2 is 15, 3→1 is 12.
   So choose 3→1 because it's smaller.

6. Create child nodes, Nodes 3 and 4.
   (a) Right child (Node 4): label is 3→1.
      Bound is 48.
      Incoming matrix for Node 4:

   1 2 4
   M 0 0
   O 0 0
   3 3
   \[ L = 48. \]

   (b) Left child (Node 3): label is 3→1.
      Bound is 48+3 = 51.
      Incoming matrix for Node 3:

   1 2 4
   M 0 0
   O M 0
   3 N 3 3
   \[ L = 48. \]

7. Done processing Node 0.

---

Process Node 2 next, because it has
the smallest bound.

Process Node 2:

Incoming matrix:

1 2 4
M 3 0
O M 0
3 0 6 3
\[ L = 45. \]

1. Compute opportunity matrix.
   (a) Set L = 45.
   (b) Ensure every row and column
       has exactly one M:

   1 2 4
   M 3 0
   O M 0
   3 0 6 3

5. Choose largest regret. (continued) →
Process Node 4 next, because it has the smallest bound.

Process Node 4:

Incoming matrix: \[
\begin{array}{ccc}
1 & 2 & 4 \\
2 & 0 & 0 \\
0 & M & 0 \\
\end{array}
\] \quad L = 48.

1. Compute opportunity matrix.
   (a) Set \( L = 48 \).
   (b) Ensure every row and column has exactly one \( M \):
   \[
   \begin{array}{ccc}
   2 & 4 \\
   1 & M & \leftarrow \text{M here.} \\
   2 & M & 0 \\
   \end{array}
   \] \quad \text{(L does not change.)}
   (c) Every column has at least one zero: \( \checkmark \)
   (d) Every row has at least one zero: \( \checkmark \)
   So the matrix above is the opportunity matrix for Node 4. \( L \)-value is 48.

2. First special case applies, because opportunity matrix is 2\( \times \)2.
   (a) Opportunity matrix is \[
   \begin{bmatrix}
   0 & M \\
   M & 0 \\
   \end{bmatrix}
   \]
   (b) Create one child node, Node 5. Label is \( 1 \rightarrow 2, \ 2 \rightarrow 4 \)
   (these are the links corresponding to zeroes in opportunity matrix).
   Bound is 48 (\( L \)-value of 2\( \times \)2 opportunity matrix).
   (c) Child node represents a complete circuit:
   \[
   \begin{align*}
   4 & \rightarrow 3 \\
   3 & \rightarrow 1 \\
   1 & \rightarrow 2 \\
   2 & \rightarrow 4 \\
   \end{align*}
   \]
   i.e., \[
   \begin{array}{c}
   1 \\
   2 \\
   3 \\
   \end{array}
   \quad \begin{array}{c}
   \rightarrow \\
   \rightarrow \\
   \rightarrow \\
   \end{array}
   \]
   \[
   \begin{array}{c}
   4 \\
   \end{array}
   \quad \begin{array}{c}
   \leftarrow \\
   \leftarrow \\
   \leftarrow \\
   \end{array}
   \]
   Distances from original distance matrix
   \[
   \begin{array}{c}
   15 \\
   4 \\
   7 \\
   \end{array}
   \quad \begin{array}{c}
   \rightarrow \\
   \rightarrow \\
   \rightarrow \\
   \end{array}
   \]
   \[
   \begin{array}{c}
   2 \\
   \end{array}
   \quad \begin{array}{c}
   \leftarrow \\
   \leftarrow \\
   \leftarrow \\
   \end{array}
   \]
   \[
   \begin{array}{c}
   15+14+7+12=48. \\
   \end{array}
   \]
   (d) Check that total length of circuit equals child node bound: \( 15+14+7+12=48 \). 
   (e) Done processing Node 4. Skip steps 3 through 7.

Now, a complete circuit has been found. Is it optimal?

Yes, because there is no unexplored non-terminal node with a smaller
bound than the length of this circuit (both 49 and 51 are greater than 48).

So stop: The circuit above is optimal.