1. Chinese Postman algorithm

A.

(1) Three sets of nodes

<table>
<thead>
<tr>
<th></th>
<th>$S_{I=0}$</th>
<th>$S_{I&lt;0}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_{I&gt;0}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1(2:1)</td>
<td>3(1:1)</td>
<td>2(1:2)</td>
</tr>
<tr>
<td>4(2:1)</td>
<td>6(2:2)</td>
<td>5(1:2)</td>
</tr>
</tbody>
</table>

(2) MDT  1 $\rightarrow$ 2   4 $\rightarrow$ 5

B. Cycles
Cycle1: 1-A-5-I-6-C-1
Cycle 2: 6-E-2-D-3-F-4-G-6

Cycle 3: 1-A-5-I-6-C-1

Cycle 4: 1-A-5-H-4-G-6-E-2-B-1

C, Join Cycles
- Join Cycle 1 and Cycle 2 at node 6
  Cycle 1,2: 1-A-5-I-6-E-2-D-3-F-4-G-6-C-1
- Join Cycle 1,2 and Cycle 3 at node 5
  Cycle 1,2,3: 1-A-5-I-6-C-1-A-5-I-6-E-2-D-3-F-4-G-6-C-1
- Join Cycle 1,2,3 and Cycle 4 at node 1
  Cycle 1,2,3,4 i.e. the Postman Tour:
  1-A-5-I-6-C-1-A-5-H-4-G-6-E-2-B-1-A-5-I-6-E-2-D-3-F-4-G-6-C-1

D, Weight of the Postman Tour

\[ \text{weight} = 3 \times w_A + 2 \times (w_I + w_E + w_C + w_G) + w_B + w_D + w_F + w_H \]
\[ = 3 \times 1 + 2 \times (1 + 1 + 2 + 3) + 3 + 2 + 3 + 2 \]
\[ = 27 \]