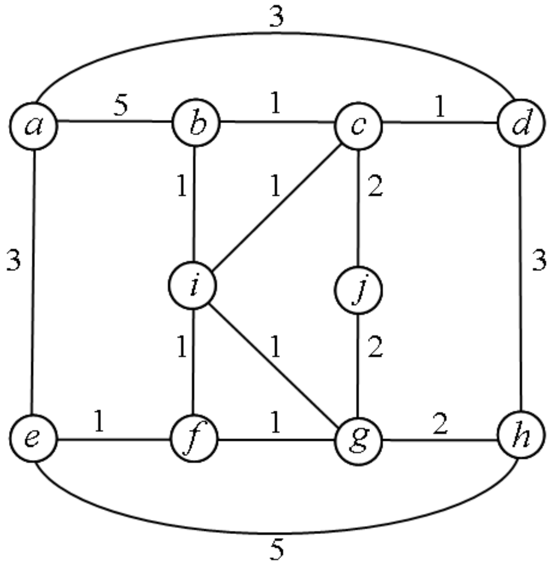
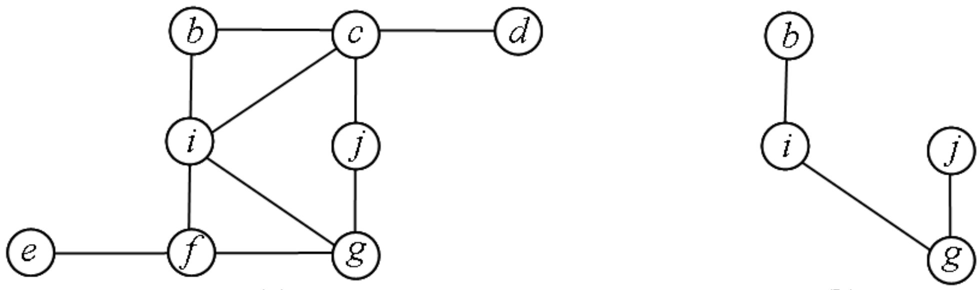
Solutions to question set for Chapter 11

All the questions in this question set concern the following graph G:



1. Prove that $\nu(G) < \tau(G)$, that is, prove that a minimum cover of this graph is strictly bigger than a maximum matching.

   SOLUTION: It is easy to see that G has a perfect matching (see Q. 4). This will have size 5. So we must show that no cover can have size 5. Since $\nu(G) \le \tau(G)$ this will confirm the strict inequality. Thus, let C be a minimum cover. Observe that the cycle *adhea* must have two non-adjacent vertices in C since no other vertices of G can cover these four edges. Suppose, without loss of generality, that $a, h \in C$. The remaining vertices of C must cover the edges of the subgraph below (a):



(a) **(b)**

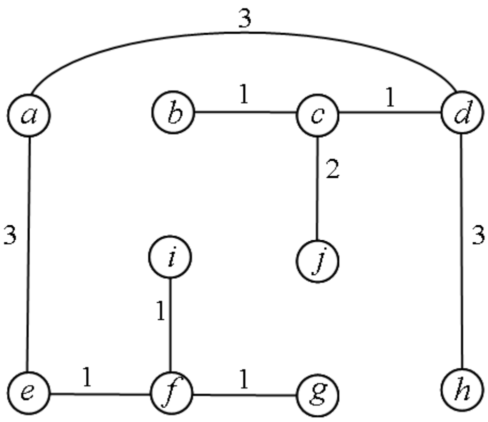
   We must cover cd and ef. Since the other edges adjacent to d and e are already covered C cannot be smaller than is obtained by adding c and f to C. This leaves the subgraph at (b). The edges of this subgraph cannot be covered by a single vertex, so $|C| \ge 6$.

2. Write down for the graph G the output of Dijkstra's algorithm, using

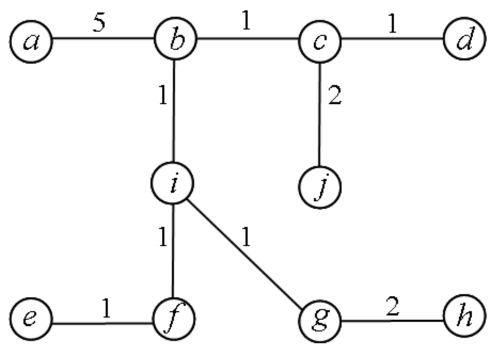
   (i) root vertex a; (ii) root vertex b.

   You need not show the steps the algorithm takes but you should show the final output tree and give a table of distances from the root vertex.

   SOLUTION: The trees are shown below at (a) and (b) respectively:



(a)



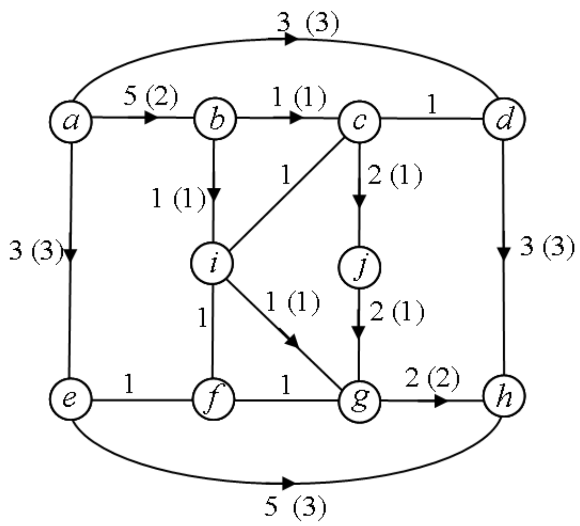
(b)

The tables of distances are:

v	a	b	c	d	e	f	g	h	i	j
dGav	0	5	4	3	3	4	5	6	5	6
dGbv	5	0	1	2	3	2	2	4	1	3

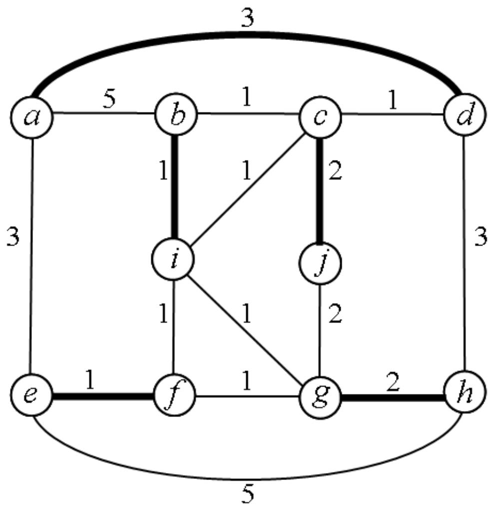
3. If the labels on the edges of G are taken to be capacities, and each edge may be traversed in either direction, give a maximum flow from a to h in G. You need not show your working.

   SOLUTION: We can send 3 units of flow around the 'outside' edges of the graph in each direction. Then we send a further 2 units through the 'inside edges'. The result is a flow of value 8 as shown below (the edges without arrows have not been used so the flow value is zero in these edges):

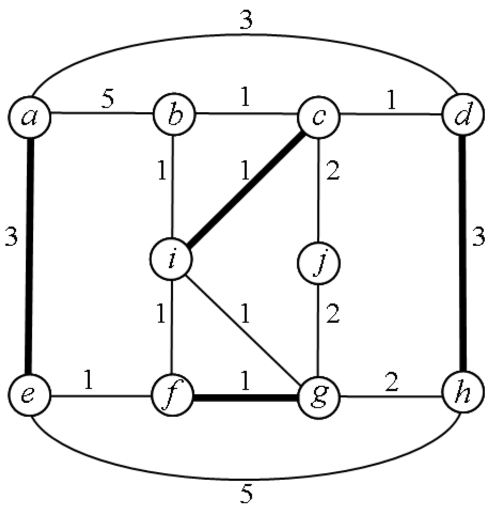


4. Find a perfect matching for the graph G and a matching which is maximal but not perfect. Find also a perfect matching of minimum total weight.

   SOLUTION: A perfect matching is shown below at (a). This also has minimum total weight among perfect matchings. A maximal matching which is not perfect is shown at (b) (this cannot be extended since every edge has at least one saturated endpoint).



(a)

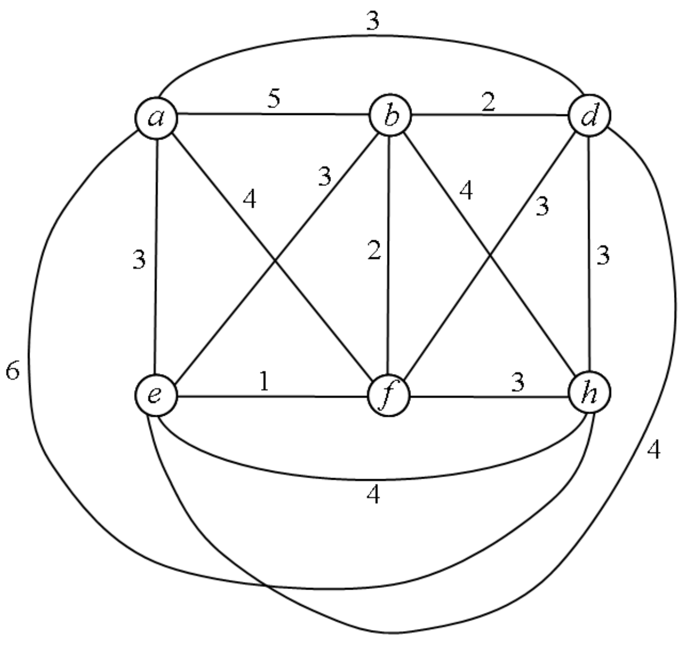


(b)

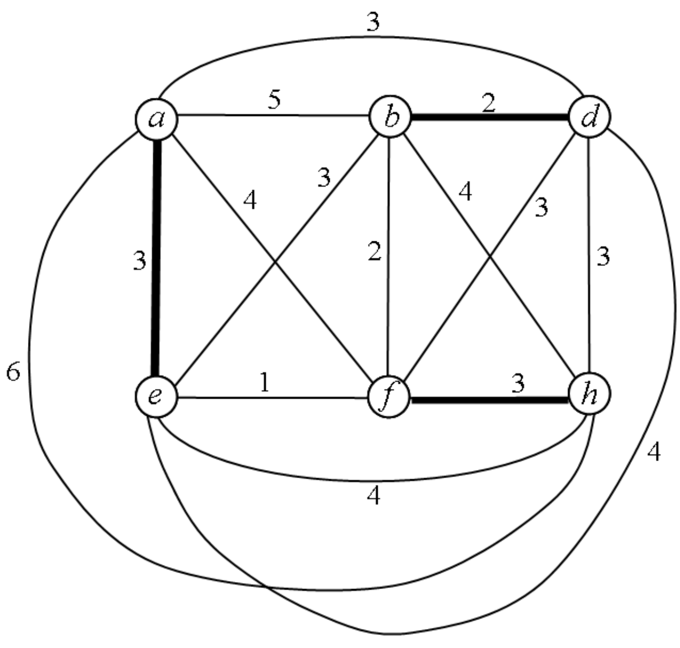
5. Run the Chinese Postman algorithm for the graph G. You need not show how the Recursive Euler Tour algorithm is used but you should show how the graph H is constructed and used to minimise your final tour of the graph.

   SOLUTION: The vertices of odd degree are a, b, d, e, f, g. The graph H on these vertices has an edge joining every pair of vertices. The weights on the edges incident with a and b are obtained from the table in the solution to question 2. The other shortest distances between vertices in G are obtained by inspection of the graph G. The result is shown below in (a).

   In (b) a minimum-weight matching for H is shown. Not that this is not the only matching of weight 8, others are: $\{ae, bf, dh\}$ and $\{ad, bh, ef\}$.

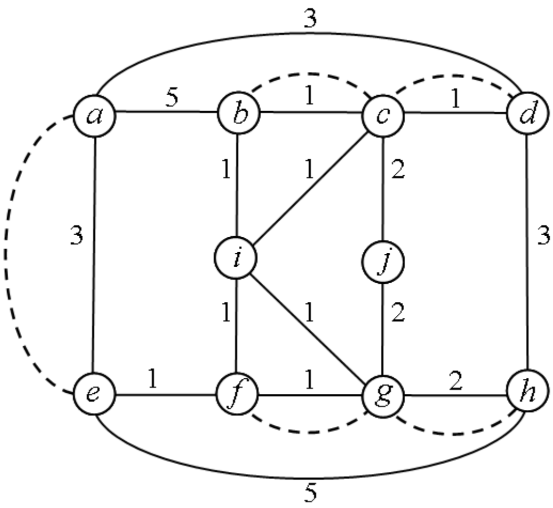


(a)



(b)

The matching in H gives paths between odd-degree vertices in G as shown:



The resulting graph has 21 edges. So a 21-edge tour visiting each edge will be an Euler tour of this graph and a minimum Chinese Postman solution for G:

$$a\, e\, a\, b\, c\, b\, i\, c\, d\, c\, j\, g\, h\, g\, f\, g\, i\, f\, e\, h\, d\, a.$$