Theorem 5.1.9 [Characterization of 2-Connected Graphs]. *Let G be a connected graph with at least three vertices. Then the following statements are equivalent.*

1. *Graph G is 2-connected.*
2. *For any two vertices of G, there is a cycle containing both.*
3. *For any vertex and any edge of G, there is a cycle containing both.*
4. *For any two edges of G, there is a cycle containing both.*
5. *For any two vertices and one edge of G, there is a path containing all three.*
6. *For any three distinct vertices of G, there is a path containing all three.*
7. *For any three distinct vertices of G, there is a path containing any two of them which does not contain the third.* ◊

EXERCISES for Section 5.1

5.1.1[S] Find the other two 2-element vertex-cuts in the graph of Example 5.1.1.

In Exercises 5.1.2 through 5.1.5, either draw a graph meeting the specifications or explain why no such graph exists.

5.1.2 A 6-vertex graph G such that $\kappa_v(G) = 2$ and $\kappa_e(G) = 2$.

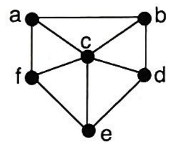
5.1.3 A connected graph with 11 vertices and 10 edges and no cut-vertices.

5.1.4 A 3-connected graph with exactly one bridge.

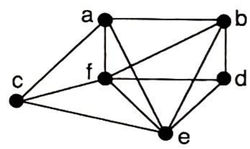
5.1.5[S] A 2-connected 8-vertex graph with exactly two bridges.

In Exercises 5.1.6 through 5.1.11, determine the vertex- and edge-connectivity of the given graph.

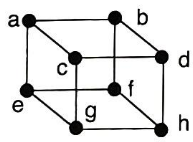
5.1.6



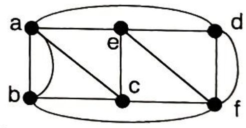
5.1.7



5.1.8



5.1.9[S]



5.1.10 Complete bipartite graph $K_{4,7}$.

5.1.11 Cube graph Q_4.

5.1.12 Determine the vertex- and edge-connectivity of the complete bipartite graph $K_{m,n}$.

5.1.13[S] Determine the vertex-connectivity and edge-connectivity of the Petersen graph (§1.2), and justify your answer. (Hint: Use the graph's symmetry to reduce the number of cases to consider.)

In Exercises 5.1.14 through 5.1.17, give an example of a graph G satisfying the given conditions.

5.1.14 $\kappa_v(G) = \kappa_e(G) = \delta_{min}(G)$

5.1.15 $\kappa_v(G) = \kappa_e(G) < \delta_{min}(G)$

5.1.16 $\kappa_v(G) < \kappa_e(G) = \delta_{min}(G)$

5.1.17[S] $\kappa_v(G) < \kappa_e(G) < \delta_{min}(G)$

5.1.18 Let $v_1, v_2, \ldots, v_k$ be k distinct vertices of a k-connected graph G, and let G^w be the graph formed from G by joining a new vertex w to each of the v_i's. Show that $\kappa_v(G^w) = k$.

5.1.19 Let G be a k-connected graph, and let v be a vertex not in G. Prove that the suspension $H = G + v$ (§2.4) is $(k+1)$-connected.

5.1.20 Prove that there exists no 3-connected simple graph with exactly seven edges.

5.1.21^S Let a, b, and c be positive integers with $a \leq b \leq c$. Show that there exists a graph G with $\kappa_v(G) = a$, $\kappa_e(G) = b$, $\delta_{min}(G) = c$.

DEFINITION: A ***unicyclic*** graph is a connected graph with exactly one cycle.

5.1.22 Show that the edge-connectivity of a unicyclic graph is no greater than 2.

5.1.23^S Characterize those unicyclic graphs whose vertex-connectivity equals 2.

5.1.24 Prove the characterization of 2-connected graphs given by Theorem 5.1.0.

5.1.25 Prove that if G is a connected graph, then $\kappa_v(G) = 1 + min_{v \in V}\{\kappa_v(G - v)\}$.

5.1.26^S Find a lower bound on the number of vertices in a k-connected graph with diameter d (§1.4) and a graph that achieves that lower bound (thereby showing that the lower bound is *sharp*).

5.2 CONSTRUCTING RELIABLE NETWORKS

In this section we examine methods for constructing graphs with a prescribed vertex-connectivity. In light of earlier remarks, these graphs amount to blueprints for reliable networks.

Whitney's Synthesis of 2-Connected Graphs and 2-Edge-Connected Graphs

DEFINITION: A ***path addition*** to a graph G is the addition to G of a path between two existing vertices of G, such that the edges and internal vertices of the path are not in G. A ***cycle addition*** is the addition to G of a cycle that has exactly one vertex in common with G.

DEFINITION: A ***Whitney-Robbins synthesis*** of a graph G from a graph H is a sequence of graphs, $G_0, G_1, \ldots, G_l$, where $G_0 = H$, $G_l = G$, and G_i is the result of a path or cycle addition to G_{i-1}, for $i = 1, \ldots, l$. If each G_i is the result of a path addition *only*, then the sequence is called a ***Whitney synthesis***.

Example 5.2.1: Figure 5.2.1 shows a 4-step Whitney synthesis of the cube graph Q_3, starting from the cycle graph C_4.

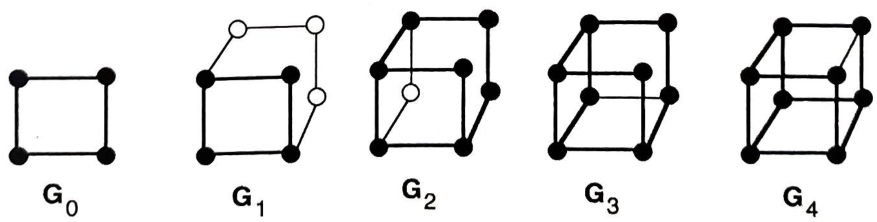
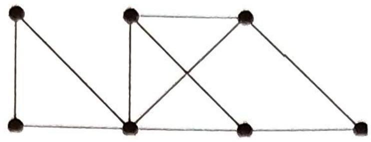


Figure 5.2.1 A Whitney synthesis of the cube graph Q_3.

5.2.4 For the 2-edge-connected graph shown, find a 3-step Whitney-Robbins synthesis from a cycle.



5.2.5^S Find a 1-step Tutte synthesis of the complete bipartite graph $K_{3,3}$.

$\boxed{5.2.6}$ Use Tutte Synthesis Theorem 5.2.5 to show that the cube graph Q_3 is 3-connected.

5.2.7^S Use Tutte Synthesis Theorem 5.2.5 to show that the Harary graph $H_{3,n}$ is 3-connected for all $n \geq 4$.

5.2.8 Show that the number of edges in the Harary graph $H_{2r,n}$ is rn.

In Exercises 5.2.9 through 5.2.12, use Algorithm 5.2.1 to construct the specified Harary graph.

5.2.9 $H_{4,7}$

$\boxed{5.2.10}$ $H_{5,7}$

5.2.11^S $H_{6,8}$

5.2.12 $H_{6,9}$

5.2.13 Show that Algorithm 5.2.1 correctly handles the three cases of the Harary construction.

5.2.14^S Prove Corollary 5.2.8.

$\boxed{5.2.15}$ Prove Case 3 of Theorem 5.2.7.

5.2.16 [*Computer Project*] Write a computer program whose inputs are two positive integers k and n, with $k < n$, and whose output is the adjacency matrix of the Harary graph $H_{k,n}$ with the vertices labeled $0, 1, \ldots, n-1$.

5.3 MAX-MIN DUALITY AND MENGER'S THEOREMS

Borrowing from operations research terminology, we consider certain *primal-dual pairs* of optimization problems that are intimately related. Usually, one of these problems involves the maximization of some objective function, while the other is a minimization problem. A feasible solution to one of the problems provides a bound for the optimal value of the other problem (this is sometimes referred to as *weak duality*), and the optimal value of one problem is equal to the optimal value of the other (*strong duality*). Menger's Theorems and their many variations epitomize this primal-dual relationship. The following terminology is used throughout this section.

u,v 在 G-S 中不同的 component

DEFINITION: Let u and v be distinct vertices in a connected graph G. A vertex subset (or edge subset) S is ***u-v separating*** (or ***separates*** u and v), if the vertices u and v lie in different components of the deletion subgraph $G - S$.

Thus, a u-v separating vertex set is a vertex-cut, and a u-v separating edge set is an edge-cut. When the context is clear, the term ***u-v separating set*** will refer either to a u-v separating vertex set or to a u-v separating edge set.

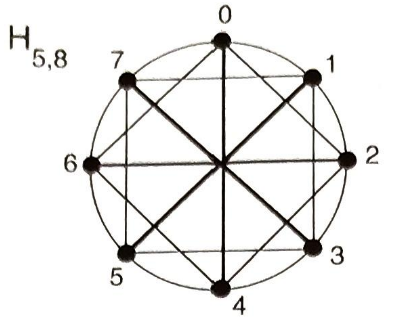


Figure 5.2.8 Harary graph $H_{5,8}$ is $H_{4,8}$ plus four diameters.

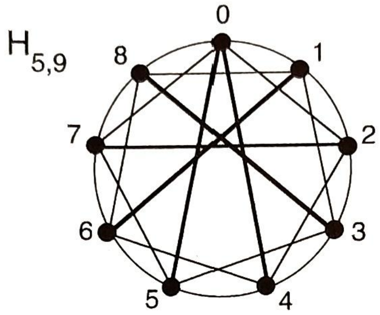


Figure 5.2.9 Harary graph $H_{5,9}$ is $H_{4,9}$ plus five quasi-diameters.

Remark: An algorithm that realizes the Harary construction appears at the end of this section.

Theorem 5.2.7. *The Harary graph $H_{k,n}$ is k-connected.*

Proof: *Case 1:* $k = 2r$

Suppose that $2r-1$ vertices $i_1, i_2, \ldots, i_{2r-1}$ are deleted from graph $H_{2r,n}$, and let x and y be any two of the remaining vertices. It suffices to show that there is an x-y path in the vertex-deletion subgraph $H_{2r,n} - \{i_1, i_2, \ldots, i_{2r-1}\}$.

The vertices x and y divide the perimeter of the original n-cycle into two sectors. One of these sectors contains the clockwise sequence of vertices from x to y, and the other sector contains the clockwise sequence from y to x. From one of these sequences, no more than $r-1$ of the vertices $i_1, i_2, \ldots, i_{2r-1}$ were deleted. Assume, without loss of generality, that this is the sector that extends clockwise from x to y, and let S be the subsequence of vertices that remain between x and y in that sector after the deletions. The gap created between two consecutive vertices in subsequence S is the largest possible when the $r-1$ deleted vertices are consecutively numbered, say, from a to $a+r-2$, using addition modulo n (see Figure 5.2.10 below). But even in this extreme case, the resulting gap is no bigger than r. That is, $|j-i|_n \leq r$, for any two consecutive vertices i and j in subsequence S.

Thus, by the construction, every pair of consecutive vertices in S is joined by an edge. Therefore, subsequence S is the vertex sequence of an x-y path in $H_{2r,n} - \{i_1, i_2, \ldots, i_{2r-1}\}$. ◊ (Case 1)

Case 2: $k = 2r+1$ and n even

Suppose that D is a set of $2r$ vertices that are deleted from graph $H_{2r+1,n}$, and let x and y be any two of the remaining vertices. It suffices to show that there is an x-y path in the vertex-deletion subgraph $H_{2r+1,n} - D$.

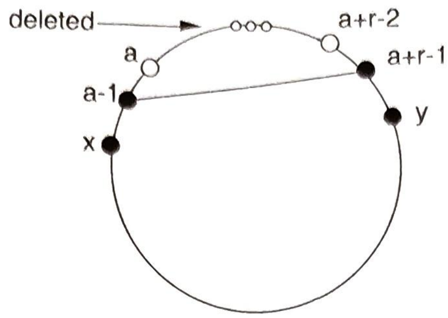


Figure 5.2.10

If one of the sectors contains fewer than r of the deleted vertices, then, as in Case 1, the subsequence of vertices left in that sector forms a path between vertices x and y. Therefore, assume both of the sectors contain exactly r of the deleted vertices. Further assume that both sets of deleted vertices are consecutively numbered, thereby creating the largest possible gap between consecutive vertices in each of the remaining subsequences.

Using addition modulo n, let $A = \{a, a+1, \ldots, a+r-1\}$ be the set of vertices that were deleted from one sector, and let $B = \{b, b+1, \ldots, b+r-1\}$ be the other set of deleted vertices. Each of the two subsets of remaining vertices is also consecutively numbered. One of these subsets starts with vertex $a+r$ and ends with vertex $b-1$, and the other starts with $b+r$ and ends with $a-1$ (see Figure 5.2.11).

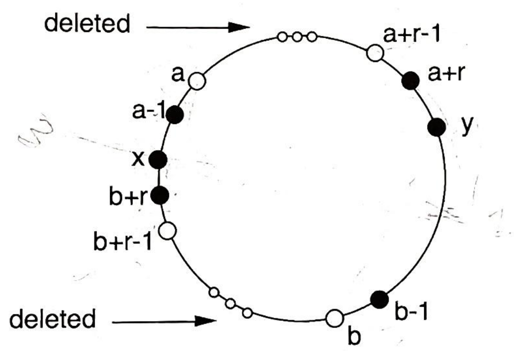


Figure 5.2.11 **The four subsets of consecutively numbered vertices.**

Let $l = |(b+r)-(a-1)|_n$ and let $w = (b+r) + \lfloor \frac{l}{2} \rfloor$. Then w is "halfway" between vertices $b+r$ and $a-1$, moving clockwise from $b+r$ to $a-1$. Now let $z = w + \frac{n}{2}$. Then z is halfway between $b+r$ and $a-1$, moving counterclockwise from $b+r$ to $a-1$. But both subsets of deleted vertices are of equal size r, which implies that z is in the other subset of remaining vertices (see Figure 5.2.12). Moreover, since $|z-w|_n = \frac{n}{2}$, there is an edge joining w and z, by the definition of $H_{2r+1,n}$ for Case 2.

Finally, there is a path from w to x, since both vertices are on a sector of the n-cycle left intact by the vertex deletions. By the same argument, there is a path from z to y. These two paths, together with the edge between w and z, show that there is an x-y path in the vertex-deletion subgraph $H_{2r+1,n} - D$. ♢ (Case 2)

Case 3: $k = 2r+1$ and n odd

The argument for Case 3 is similar to the one used for Case 2. ♢ (Exercises)

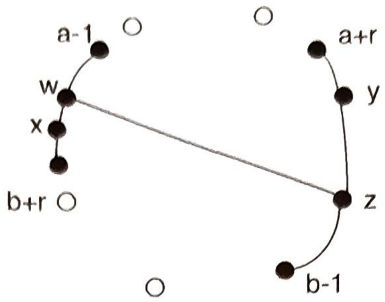


Figure 5.2.12 A diameter connecting the two remaining sectors.

The following result states that the Harary graphs are also optimal with respect to achieving a given edge-connectivity with a minimum number of edges.

Corollary 5.2.8. *The Harary graph $H(k,n)$ is a k-edge-connected, n-vertex graph with the fewest possible edges.* ◇ *(Exercises)*

Algorithm 5.2.1: Constructing an Optimal k-Connected n-Vertex Graph

Input: positive integers k and n, with $k < n$.
Output: Harary graph $H_{k,n}$ with the standard 0-based labeling.

```
Initialize graph H to be n isolated vertices, with labels 0, ..., n.
Let r = ⌊k/2⌋.
For i = 0 to n − 2
    For j = i + 1 to n − 1
        If j − i ≤ r OR n + i − j ≤ r
            Create an edge between vertices i and j.
[This completes the construction of H_{2r,n}.]
If k is even
    Return graph H.
Else
    If n is even
        For i = 0 to n/2 − 1
            Create an edge between vertex i and vertex (i + n/2)
    Else
        Create an edge from vertex 0 to vertex (n−1)/2.
        Create an edge from vertex 0 to vertex (n+1)/2.
        For i = 1 to (n−3)/2
            Create an edge between vertex i and vertex (i + (n+1)/2).
Return graph H.
```

EXERCISES for Section 5.2

5.2.1[S] Find a Whitney synthesis of the 7-vertex wheelgraph W_6.

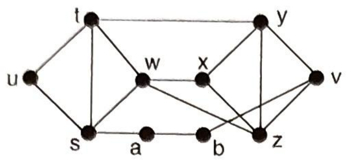
5.2.2 Find a Whitney synthesis of the complete graph K_5.

5.2.3 Find a 3-step Whitney synthesis of the cube graph Q_3, starting with a 4-cycle.

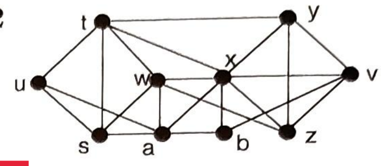
EXERCISES for Section 5.3

In Exercises 5.3.1 through 5.3.4, find the maximum number of internally disjoint u-v paths for the given graph, and use Certificate of Optimality (Corollary 5.3.3) to justify your answer.

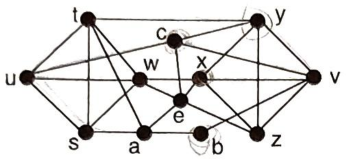
5.3.1[S]



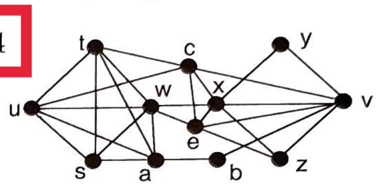
5.3.2



5.3.3



5.3.4



In Exercises 5.3.5 through 5.3.8, find the maximum number of edge-disjoint u-v paths for the specified graph, and use Edge Form of Certificate of Optimality (Proposition 5.3.9) to justify your answer.

5.3.5[S] The graph of Exercise 5.3.1.

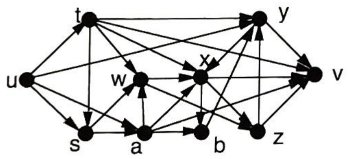
5.3.6 The graph of Exercise 5.3.2.

5.3.7 The graph of Exercise 5.3.3.

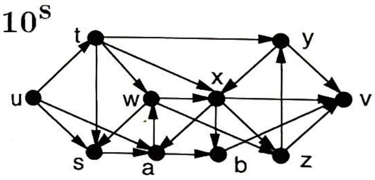
5.3.8 The graph of Exercise 5.3.4.

In Exercises 5.3.9 and 5.3.10, find the maximum number of arc-disjoint directed u-v paths for the given graph, and use the digraph version of Proposition 5.3.9 to justify your answer. That is, find k arc-disjoint u-v paths and a set of k arcs that separate vertices u and v, for some integer k.

5.3.9



5.3.10[S]



5.3.11 Prove Certificate of Optimality (Corollary 5.3.3).

5.3.12 Prove Edge Form of Certificate of Optimality (Proposition 5.3.9).

5.3.13 Prove Edge Form of Menger's Theorem 5.3.10.

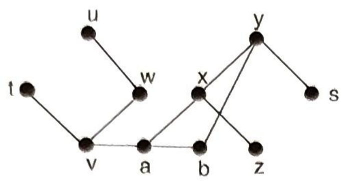
5.3.14 Prove Whitney's k-Edge-Connected Characterization Theorem 5.3.11.

5.3.15[S] Prove that Tutte Synthesis Theorem 5.2.5 always produces a 3-connected graph. (Hint: Develop an induction that uses Whitney's k-Connected Characterization Theorem 5.3.6.)

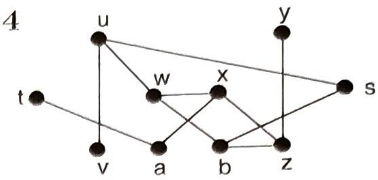
5.3.16 Let G be a simple k-connected graph with $k \geq 2$. Let S be a set of two edges and W a set of $k-2$ vertices. Prove that there exists a cycle in G containing the elements of S and W.

5.3.17 Let G be a simple k-connected graph. Let $W = \{w_1, w_2, \ldots, w_k\}$ be a set of k vertices and $v \in V_G - W$. Prove that there exists a v-w_i path P_i, $i = 1, \ldots, k$, such that the collection $\{P_i\}$ of k paths is internally disjoint.

5.4.3



5.4.4



5.4.5[S] Draw a graph whose block graph is the complete graph K_3.

5.4.6 Find two non-isomorphic connected graphs with six vertices, six edges, and three blocks.

5.4.7 Find a graph whose block graph is the n-cycle graph C_n.

5.4.8[S] How many non-isomorphic simple connected graphs are there that have seven vertices, seven edges, and three blocks?

In Exercises 5.4.9 through 5.4.12, apply Algorithm 5.4.1 to the specified graph.

5.4.9 The graph of Exercise 5.4.1. **5.4.10** The graph of Exercise 5.4.2.
5.4.11 The graph of Exercise 5.4.3. **5.4.12** The graph of Exercise 5.4.4.

5.4.13 Prove or disprove: Every simple graph is the block graph of some graph.

5.4.14[S] Prove or disprove: Two graphs are isomorphic if and only if their block graphs are isomorphic.

5.4.15 Prove Corollary 5.4.3.

5.4.16 Prove Corollary 5.4.4.

5.4.17 Prove Corollary 5.4.5.

DEFINITION: Let G be a simple connected graph with at least two blocks. The ***block-cutpoint graph*** $bc(G)$ of G is the bipartite graph with vertex bipartition $\langle V_b, V_c \rangle$, where the vertices in V_b bijectively correspond to the blocks of G and the vertices in V_c bijectively correspond to the cut-vertices of G, and where vertex v_b is adjacent to vertex v_c if cut-vertex c is a vertex of block b.

In Exercises 5.4.18 through 5.4.21, draw the block-cutpoint graph $bc(G)$ of the specified graph G, and identify the vertices in $bc(G)$ that correspond to leaf blocks of G.

5.4.18[S] The graph of Exercise 5.4.1. **5.4.19** The graph of Exercise 5.4.2.
5.4.20 The graph of Exercise 5.4.3. **5.4.21** The graph of Exercise 5.4.4.

5.4.22[S] Let G be a simple connected graph with at least two blocks. Prove that the block-cutpoint graph $bc(G)$ is a tree.

5.4.23 Prove Proposition 5.4.6. (Hint: See Exercise 5.4.22.)

5.4.24 [*Computer Project*] Implement Algorithm 5.4.1 and run the program, using each of the graphs in Exercises 5.4.-2 through 5.4.1 as input.

5.5 SUPPLEMENTARY EXERCISES

5.5.1 Calculate the vertex-connectivity of $K_{4,7}$.

5.5.2 Prove that the complete bipartite graph $K_{m,m}$ is m-connected.

5.5.3 Prove that the vertex-connectivity of the hypercube graph Q_n is n.

Algorithm 5.4.1: Block-Finding

Input: a connected graph G.
Output: the vertex-sets $B_1, B_2, \ldots, B_l$ of the blocks of G.

Apply Algorithm 4.4.3 to find the set K of cut-vertices of graph G.
Initialize the block counter $i := 0$.
For each cut-vertex v in set K (in order of decreasing $dfnumber$)
    For each child w of v in depth-first search tree T
        If $low(w) \geq dfnumber(v)$
            Let T^w be the subtree of T rooted at w.
            $i := i + 1$
            $B_i := V_{T^w} \cup \{v\}$
            $T := T - V_{T^w}$
Return sets $B_1, B_2, \ldots, B_i$.

COMPUTATIONAL NOTE: With some relatively minor modifications of Algorithm 5.4.1, the cut-vertices and blocks of a graph can be found in one pass of a depth-first search (see e.g., [AhHoUl83], [Ba83], [ThSw92]). A similar one-pass algorithm for finding the strongly connected components of a digraph is given in §12.5.

Block Decomposition of Graphs With Self-Loops

In a graph with self-loops, each self-loop and its endpoint are regarded as a distinct block, isomorphic to the bouquet B_1. The other blocks of such a graph are exactly the same as if the self-loops were not present. This extended concept of block decomposition preserves the property that the blocks partition the edge-set.

Example 5.4.3: The block decomposition of the graph G shown in Figure 5.4.4 contains five blocks, three of which are self-loops.

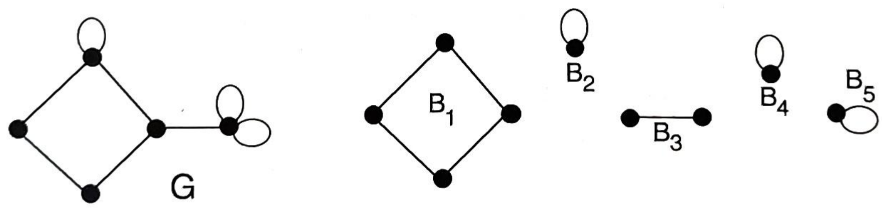
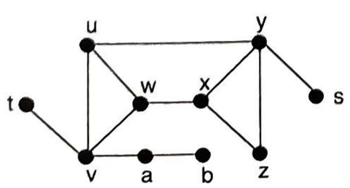


Figure 5.4.4 **A graph G and its five blocks.**

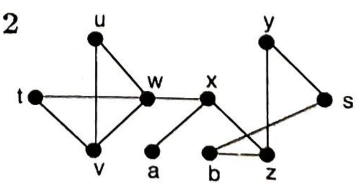
EXERCISES for Section 5.4

In Exercises 5.4.1 through 5.4.4, identify the blocks in the given graph and draw the block graph.

5.4.1[S]



5.4.2



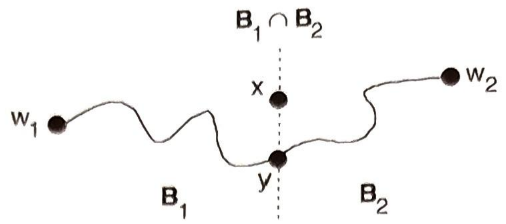


Figure 5.4.2 Two blocks cannot have more than one vertex in common.

The following assertions are immediate consequences of Proposition 5.4.2.

Corollary 5.4.3. *The edge-sets of the blocks of a graph G partition E_G.*

◇ (Exercises)

Corollary 5.4.4. *Let x be a vertex in a graph G. Then x is a cut-vertex of G if and only if x is in two different blocks of G.* ◇ (Exercises)

Corollary 5.4.5. *Let B_1 and B_2 be distinct blocks of a connected graph G. Let y_1 and y_2 be vertices in B_1 and B_2, respectively, such that neither is a cut-vertex of G. Then vertex y_1 is not adjacent to vertex y_2.* ◇ (Exercises)

DEFINITION: The ***block graph*** of a graph G, denoted $BL(G)$, is the graph whose vertices correspond to the blocks of G, such that two vertices of $BL(G)$ are joined by a single edge whenever the corresponding blocks have a vertex in common.

Example 5.4.2: Figure 5.4.3 shows a graph G and its block graph $BL(G)$.

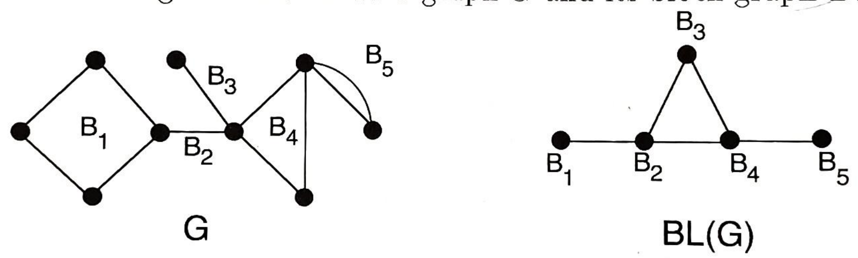


Figure 5.4.3 A graph and its block graph.

DEFINITION: A ***leaf block*** of a graph G is a block that contains exactly one cut-vertex of G.

The following result is used in §9.1 to prove *Brooks's Theorem* concerning the *chromatic number* of graph.

Proposition 5.4.6. *Let G be a connected graph with at least one cut-vertex. Then G has at least two leaf blocks.* ◇ (Exercises)

Finding the Blocks of a Graph

In §4.4, depth-first search was used to find the cut-vertices of a connected graph (Algorithm 4.4.3). The following algorithm, which uses Algorithm 4.4.3 as a subroutine, finds the blocks of a connected graph. Recall from §4.4 that $low(w)$ is the smallest *dfnumber* among all vertices in the depth-first tree that are joined to some descendant of vertex w by a non-tree edge.