San Skulrattanakulchai
Gustavus Adolphus College

Chapter 4. Bridges and Trees

Definition 1. An edge e in a connected graph G is a *bridge* if $G - e$ is disconnected. An edge e in a disconnected graph G is a *bridge* if $G' - e$ is disconnected, where G' is the connected component of G that contains e.

Definition 2. An edge e in a graph G is a *bridge* if $G - e$ has more connected components than G.

Exercise. Prove that these two definitions are equivalent.

Theorem (CZ, Theorem 4.1)**.** *An edge e of a graph G is a bridge if and only if e lies on no cycle of G.*

Proof. . . . □

Lemma (Bridge Lemma)**.** *If $e = uv$ is a bridge in a connected graph G, then $G - e$ has exactly 2 connected components, one containing u and the other containing v.*

Proof. . . . □

Definitions. An *acyclic* graph contains no cycle. A *forest* is an acyclic graph. A *tree* is a connected forest.

Lemma. *Let G be a graph. There exist vertices u and v in G with more than one $u - v$ path if and only if G contains some cycle.*

Proof. . . . □

Theorem (CZ, Theorem 4.2)**.** *A graph G is a tree if and only if every two vertices of G are connected by a unique path.*

Proof. . . . □

Theorem (CZ, Theorem 4.4)**.** *Every tree of order n has size $n - 1$.*

Proof. See class note on induction. □

Exercise 4.8 (CZ, p.92). *Prove that if every vertex of a graph G has degree at least 2, then G contains a cycle.*

Theorem (CZ, Theorem 4.3)**.** *Every nontrivial tree has at least two end vertices.*

Proof. (First Proof) Suppose T is a nontrivial tree with at most one end vertex, i.e., either T has no end vertex or it has exactly one end vertex. Since T is a nontrivial tree, none of its vertices is isolated. Therefore, if T has no end vertex, then every vertex has degree at least 2. Exercise 4.8 shows that T must contain some cycle. This contradicts the fact that T is a tree, and so acyclic. Thus, T has exactly one end vertex; let's call it x. By Theorem 4.4, Theorem 2.1, and the fact that T has one end vertex and no isolated vertex, we see that

$$\begin{aligned} 2(n-1) &= 2m \\ &= \sum_{v \in V(T)} \deg v \\ &= \deg x + \sum_{v \in V(T) \setminus \{x\}} \deg v \\ &\geq 1 + 2(n-1) \end{aligned}$$

which implies $0 \geq 1$, a contradiction.

(Second Proof) Let $P : u_0, u_1, \ldots, u_{k-1}, u_k$ be a maximal path in T. Since T is nontrivial, the length of P is at least one. Thus $u_0 \neq u_k$. We will show that both u_0 and u_k are end vertices. Consider u_0. All of its neighbors are on P because u_0 is an end vertex of the maximal path P. We know u_1 is a neighbor of u_0. In fact, it is the only neighbor. This is because if $u_j (j > 1)$ were some other neighbor of u_0, then $u_0, u_1, \ldots, u_{j-1}, u_j, u_0$ would be a cycle in T, which is impossible since T is a tree so it has no cycle. Therefore, u_0 is an end vertex as desired.

That u_k is an end vertex can be proved in a similar way. □

Corollary (CZ, Corrolary 4.6)**.** *Every forest of order n with k components has size $n-k$.*

Proof. . . . □

Definitions. Recall that a graph H is a *spanning subgraph* of a graph G if H is a subgraph of G and $V(H) = V(G)$. If T is a subgraph of G and T is a tree, we say that T is a *spanning tree of* G.

Theorem (CZ, Theorem 4.10)**.** *Every connected graph contains a spanning tree.*

Proof. Let G be a connected graph. Let T be a subgraph of G obtained from executing the following procedure.

0: $T \leftarrow G$
1: **while** T contains some cycle C **do** {
2:    let e be an edge on the cycle C
3:    $T \leftarrow T - e$
4: }
5: **return** T

We claim

1. the procedure terminates, and
2. T is a connected subgraph of G throughout the procedure.

We first prove claim 1. The graph G has a finite number m of edges. The while loop iterates by deleting an edge from the current graph as long as it contains some cycle. The empty graph on $V(G)$ is the minimal subgraph of G and it is acyclic. So the loop cannot iterate more than m times. Thus, our procedure terminates.

We now prove claim 2. T is a connected subgraph of G right after Line 0 is executed because the input graph G is assumed to be connected and Line 0 simply assigns G to T. We now show that the loop maintains connectedness of T. So suppose that T is connected before Line 1 is executed and suppose the while condition of Line 1 is true. Line 2 then picks an edge e belonging to a cycle and Line 3 deletes e from T. By Theorem 4.1, edge e is not a bridge; so deleting it from T still leaves T connected.

By claim 1, the procedure terminates. Now, it terminates only when the condition in Line 1 is false, i.e., when the current graph T contains no cycle. Thus, Lines 5 returns a subgraph T of G that is both connected and acyclic. Thus, T is a spanning tree of G. □

Theorem (CZ, Theorem 4.7)**.** *The size of every connected graph of order n is at least $n-1$.*

Proof. Let G be a connected graph of order n. By Theorem 4.10, let T be a spanning tree of G. The size of T is $n-1$ because (i) the order of G is n, and (ii) T spans G, and (iii) Theorem 4.4. Since T is a subgraph of G, the size of G is at least the size of T. Therefore, the size of G is at least $n - 1$. □

Theorem (CZ, Theorem 4.8)**.** *Let G be a graph of order n and size m. If G satisfies any two of the properties:*

1. *G is connected,*

2. *G is acyclic,*

3. *$m = n - 1$,*

then G satisfies all three properties.

Proof. ... □

Theorem (CZ, Theorem 4.9)**.** *Let T be a tree of order k. If G is a graph with $\delta(G) \geq k - 1$, then T is isomorphic to some subgraph of G.*

Proof. ... □