CS 137 - Graph Theory - Lecture 3
February 18, 2012

1.1. Review

- walk, path, cycle
- connected, disconnected

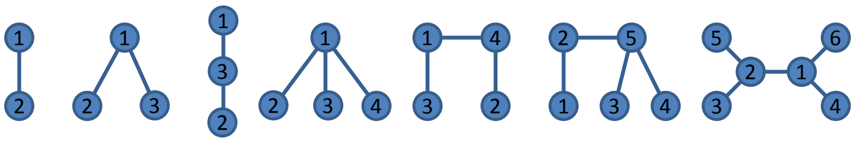
Lemma 1. *If every vertex in G has degree at least two, then G contains a cycle.*

1.2. Summary

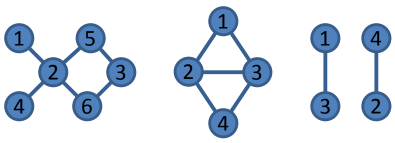
– Trees
– Prüfer's code, Cayley's formula

2. Trees

Sample trees:



Not trees:

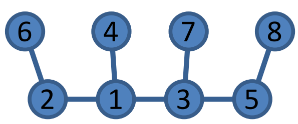


A graph G is a *tree* if it is connected and contains no cycles. A *leaf* is a vertex of degree 1.

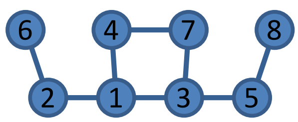
Lemma 2. *Every tree with at least 2 vertices has at least one leaf. Removing it (and the incident edge) yields again a tree.*

Proof. A consequence of Lemma 1 (use the contrapositive). □

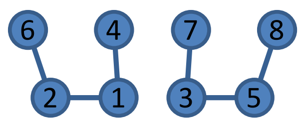
Theorem 3. *Let G be a graph with n vertices. Then the following are equivalent.*
(i) G is a tree *(G is connected and has no cycles)*
(ii) G is connected and has exactly $n-1$ edges
(iii) G has exactly $n-1$ edges and no cycles
(iv) G is minimally connected *(G is connected and removing any edge disconnects the graph)*
(v) G is maximally acyclic *(G has no cycles and adding any additional edge creates a cycle)*



a tree



adding {4,7} creates a cycle



removing {1,3} makes the graph disconnected

Proof. (i)⇒ (ii) Assume (i): G is a tree. By Lemma 2, G has a leaf u. Remove u to get a tree G'. Note that G' has $n-1$ vertices; by induction G' has $n-2$ edges and so G has $n-1$ edges.

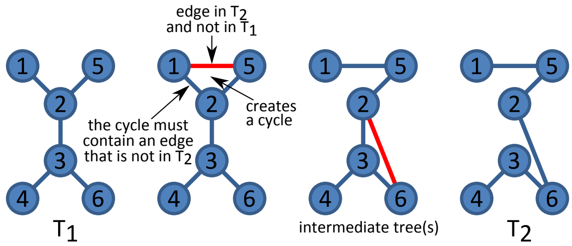
(ii)⇒ (iii) Assume (ii): G is connected, with $n-1$ edges. If G has a cycle, then remove any edge of this cycle; the resulting graph is still is connected. Repeat until no more cycles; let G' be the resulting graph; G' is connected and has no cycles, so it has $n-1$ edges (by (i)⇒(ii)), but then $G = G'$ since $n-1 = |E(G)| \geq |E(G')| = n-1$.

(iii)⇒(iv) Remove any edge. Let G' be the resulting graph. Clearly G' has no cycles since G has no cycles. If G' is connected, then G' is a tree and so it has $n-1$ edges (by (i)⇒(ii)), but then G has more than $n-1$ edges.

(iv)⇒(v) If G has a cycle, remove any edge of the cycle and the resulting graph is connected, impossible. So G has no cycle and is connected, is a tree. Add an edge $\{u, v\}$ to G; let G' be the result. Since G is connected, there is a path from u to v in G; thus G' contains a cycle.

(v)⇒ (i) Asssume (v). So G has no cycles. If (i) fails, then G is disconnected; i.e. there are vertices u, v such that there is no path from u to v in G; adding the edge $\{u, v\}$ does not create a cycle; this contradicts (v). □

Note: we can go from one tree to another by swapping edges such that all intermediate graphs are also trees.



3. Prüfer's code, Cayley's formula

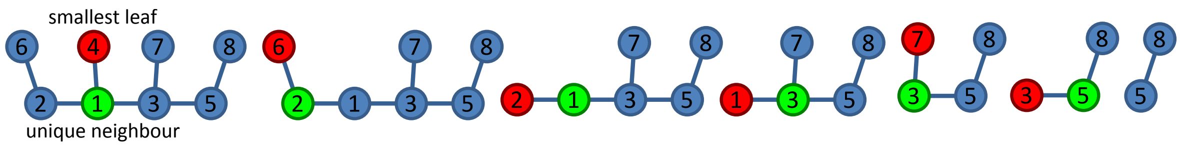
Recall that for a set A and integer n, we write $A^n = \overbrace{A \times A \times \cdots \times A}^{n \text{ times}}$. In other words, A^n denotes the set of all sequences of size n formed using elements from A. Also, recall that by Lemma 1 every tree with at least two vertices has a leaf. This is the basis of the following encoding of trees.

Algorithm: Prüfer's code

> **Input:** A tree T with vertex set $S \subseteq \mathbb{N}$ where $|S| \geq 2$.
> **Output:** A sequence $f(T) = (a_1, a_2, \ldots, a_{|S|-2}) \in S^{|S|-2}$
> **let** $T_1 = T$
> **for** each $i = 1$ to $|S| - 2$ **do**
>   **let** v be the leaf of T_i with smallest label
>   **set** a_i to be the unique neighbour of v in T
>   **construct** T_{i+1} from T_i by removing the vertex v and the edge $\{v, a_i\}$
> **end for**

Note: If $|S| = 2$, the algorithm outputs the empty sequence.

Example of a tree T for which the algorithm produces $f(T) = (1, 2, 1, 3, 3, 5)$. Reconstruction of T from the sequence $f(T)$ is shown below the theorem.



Theorem 4. *Let $S \subseteq \mathbb{N}$ with $|S| \geq 2$. There is a bijection between $S^{|S|-2}$ and the set of all trees with vertex set S.*

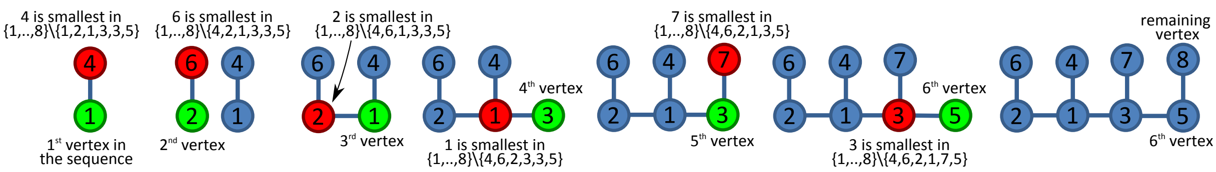
Proof. Let $n = |S|$. Consider the function f produced by the above algorithm. We show that f is the desired bijection. This will follow if we show that every sequence $(a_1, \ldots, a_{n-2}) \in S^{n-2}$ defines a unique tree T such that $f(T) = (a_1, \ldots, a_{n-2})$. If $n = 2$, then there is exactly one tree on 2 vertices and the algorithm always outputs the empty sequence, the only such sequence. So the claim clearly holds for $n = 2$.

Now assume $n > 2$ and the claim by induction holds for all sets S' of size less than n. Consider a sequence $(a_1, \ldots, a_{n-2}) \in S^{n-2}$. We need to show that $(a_1, \ldots, a_{n-2})$ can be uniquely produced by the algorithm.

Let us analyze this situation. Suppose that the algorithm produces $f(T) = (a_1, \ldots, a_{n-2})$ for some tree T. Then none of $a_1, \ldots, a_{n-2}$ is a leaf of T. Indeed, when a vertex is set to be a_i it is adjacent to a leaf in T_i. So if a_i is a leaf of T_i, then T_i has only 2 vertices. However T_i has $|S| - i + 1$ vertices, which is ≥ 3, since $i \leq |S| - 2$.

This implies that the label of the first leaf removed from (the hypothetical tree) T is precisely the minimum element of the set $S \setminus \{a_1, \ldots, a_{n-2}\}$. Let v be this element. In other words, in every tree T such that $f(T) = (a_1, \ldots, a_{n-2})$ the vertex v is a leaf whose unique neighbour is a_1.

By induction, there is a unique tree T' with vertex set $S \setminus \{v\}$ such that $f(T') = (a_2, \ldots, a_{n-2})$. Adding the vertex v and the edge $\{a_1, v\}$ to T' yields the desired unique tree T with $f(T) = (a_1, \ldots, a_{n-2})$. □



This theorem yields the following formula counting the number of labelled trees.

Theorem 5 (Cayley's formula)**.** *The number of labelled trees on n vertices is n^{n-2}.*