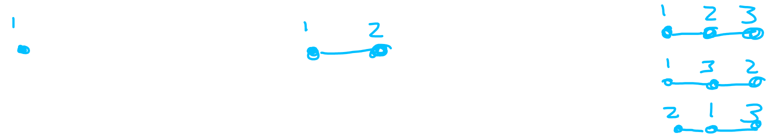
Math 38 — Graph Theory
Enumeration of trees

Nadia Lafrenière
04/22/2020

How many labeled trees are there? And up to isomorphism? This is the questions we want to ask today. We will only be able to solve one of these.

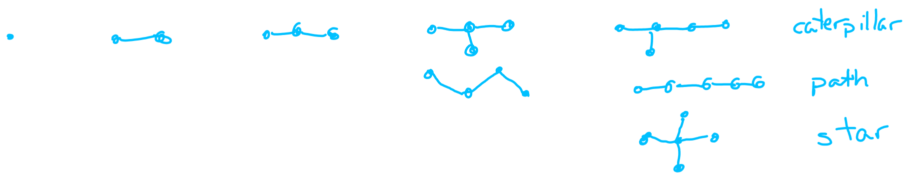
Example
The number of labeled trees with {0,1,2,3} vertices are, respectively, 1,1,1,3.

n=0, 1 tree   n=1, 1 tree   n=2, 1 tree   n=3, 3 trees



For the example with n=3, the three trees are isomorphic. However, as labeled trees they are not the same; their adjacency matrices are different (the vertex with degree 2 is not the same).

In the case of unlabeled trees, there are 1,1,1,1,2,3 trees of 0,1,2,3, 4 and 5 vertices, respectively. These are the trees up to isomorphism.



Doing the same exercise with labeled graphs, we find there are $2^{\binom{n}{2}}$ graphs. (Proof will be in homework).

So there are at most $2^{\binom{n}{2}}$ labeled trees with n vertices. But we can expect this number to be much less.

Theorem (Cayley's formula; proof is from Prüfer, 1918)
There are n^{n-2} labeled trees with n vertices, if $n \geq 1$.

Proof

This proof is done using a bijection. To do so, we have to find another collection of objects indexed by the positive integers, so that there are n^{n-2} items indexed by n. A natural choice is the n-ary sequences of length n-2. For a given tree, the generated sequence is called the Prüfer sequence or Prüfer code.

We present two algorithms: The first one takes a tree as input and transforms it into a unique n-ary sequence on length n-2; the second one takes an n-ary sequence of length n-2 and builds a unique tree with n vertices. This will prove that the number of n-ary sequences of length n-2 and the number of binary trees with the same vertices are the same.

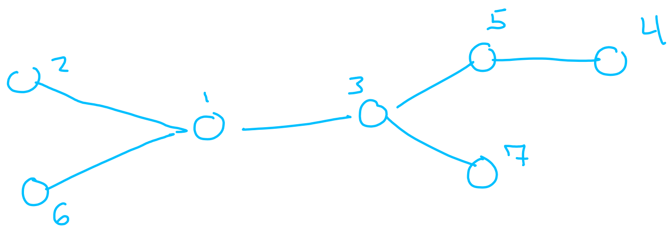
Algorithm 1: Production of the sequence

Let T be a tree with vertices $\{1,2,\dots,n\}$. If $n \leq 2$, we already checked these numbers in the example above. So assume $n \geq 2$.

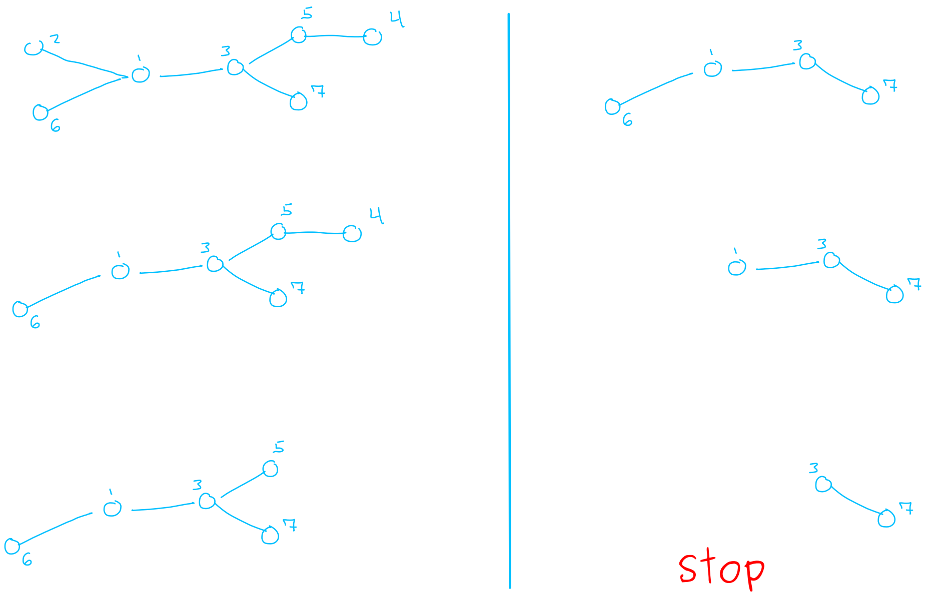
While there is more than 2 vertices, remove the leaf with smallest label (it always exists, by Lemma from 4/20). To the sequence, append the neighbor of that leaf (it is unique, since a leaf has degree 1).

Once there are only two vertices left, stop.

Example



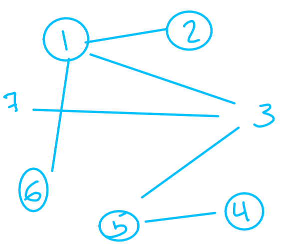
<-> (1,5,3,1,3)



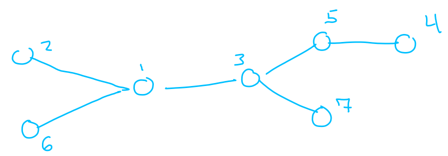
(1)
(1,5)
(1,5,3)
(1,5,3,1)
(1,5,3,1,3)

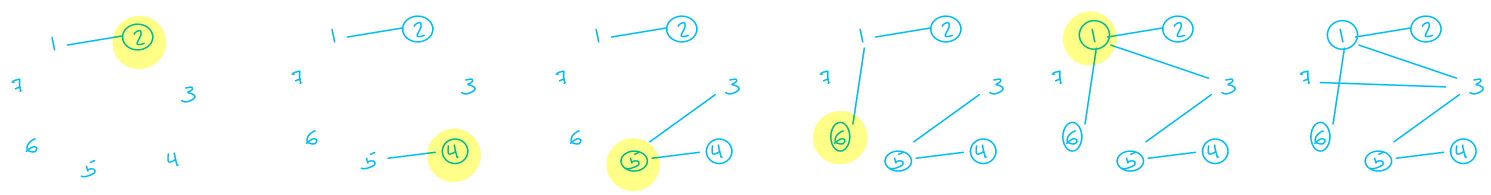
Algorithm 2: Production of the tree

Take a n-ary sequence s of length $n-2$, $n \geq 2$.
Draw n isolated vertices, and label them $\{1,2,\dots,n\}$. We will add $n-1$ edges. At the beginning, no vertex is marked.
While the sequence is not empty:
Mark the smallest unmarked vertex whose label does not appear in the sequence (this always exists, since sequence only has length $n-2$).
Delete the first element of the sequence and draw an edge between this element and the vertex you just marked. This adds one edge.
When the sequence is empty, there are $n-2$ edges, one for each element of the original sequence. There are two unmarked vertices, including one isolated. Draw an edge between them, then stop.



(1,5,3,1,3)
(5,3,1,3)
(3,1,3)
(1,3)
(3)





Claim: this is a tree. To prove this claim, we only need to prove either that it is connected or that it has no cycle, since we already know there are $n-1$ edges. We prove there is no cycle.
When we mark a vertex, we draw one last edge incident to it, and the other endpoint of that edge is an unmarked vertex. If there was a cycle, that cycle would need to have at least one edge added with both endpoints that are marked, which is not possible. That proves the claim.

Since both algorithms are well-defined, there is a bijection between n-ary sequences of length $n-2$ and the trees with n vertices.

Note that the seemingly related problem of counting unlabeled trees is much harder... to the extend that no closed formula is known to count them. The only thing we know is an asymptotic estimate of the number of trees with n vertices when n is very big; even then, the proof is very hard and requires techniques that are far beyond the scope of this class.

Counting labeled trees with a given degree sequence

From now on, we want to count trees with n vertices, labeled $\{1, \dots, n\}$, and vertex i having degree d_i. How many such trees are there? We use Prüfer sequences to solve this problem.

Observation: In the second algorithm, we always add an edge between the vertex we mark and the vertex that appears in the sequence (assuming we mark both vertices in the last step). So the degree of the vertex i is the number of occurrences of i in the sequence, plus 1.

Corollary

Given positive integers $d_1, d_2, \ldots, d_n$ summing to $2n-2$, there are exactly $\frac{(n-2)!}{\prod_i (d_i - 1)!}$ trees with vertex set $\{1, 2, \dots, n\}$ such that vertex i has degree d_i.

Proof

Using the observation, we know that the number of such trees is the number of sequences with $(d_i - 1)$ occurrences of i, for every i in $\{1, 2, \dots, n\}$. The number of sequences of length n with numbers all distinct is the number of permutations, this is $n!$ $(= 1 \times 2 \times \dots \times n)$. When a number is repeated k times, there are $k!$ fewer sequences: this is because we accounted the $k!$ permutations of these occurences of the number. Hence, the number of sequences of length $n-2$ with $d_i - 1$ occurrences of i is $\frac{(n-2)!}{\prod_i (d_i - 1)!}$

∎

Reference: Douglas B. West. Introduction to graph theory, 2nd edition, 2001. Section 2.2