Discrete Mathematics

Lecture 8

Trees

Contents

1. How to define trees
2. How to grow trees
3. How to count trees
4. How to store trees
5. The number of unlabeled trees

1 How to Define Trees

Definitions

- A **tree** is an acyclic connected graph.
- A **leaf** (叶) is a node of degree one.
- A **branch node** (内点) is of degree at least two.
- A **forest** is a disjoint union of trees.
- A **trivial tree** is on only one node.

Theorem 1.1 For $T=(V, E)$, the following statements are equivalent.

1. T is a tree, that is acyclic & connected.
2. T is connected, but deleting any edge results in a disconnected graph.
3. T has no cycles, but adding any new edge creates a cycle.
4. Each pair of nodes in T are connected by a unique path.
5. T is connected & $|E|=|V|-1$.
6. T has no cycles & $|E|=|V|-1$.

Theorem 1.1 For $T=(V, E)$, the following statements are equivalent.

1. T is a tree, that is acyclic & connected.
2. T is connected, but deleting any edge results in a disconnected graph.

(1)⇒(2)： (indirect proof)

- Suppose $\exists uv \in E$, st. $T-uv$ is connected.
- There is a path P linking u and v in $T-uv$,
- but $P+uv$ would be a cycle in T.

Theorem 1.1 For $T=(V, E)$, the following statements are equivalent.

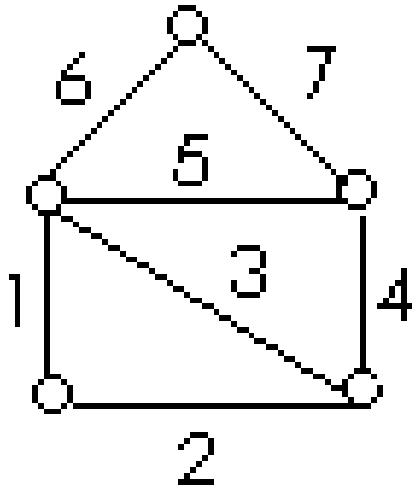
1. T is a tree, that is acyclic & connected.
2. T is connected, but deleting any edge results in a disconnected graph.

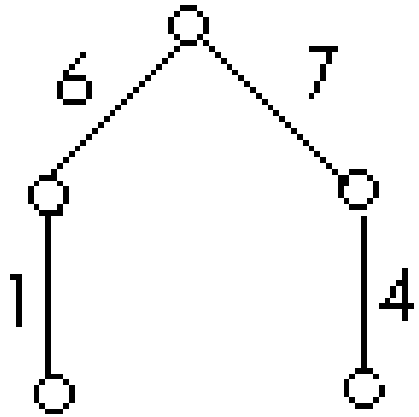
(2)⇒(1)： By Exercise 7.2.5 (b), we know if ∃a cycle $C \subset T$, then $T-e$ is still connected for all $e \in E(C)$!

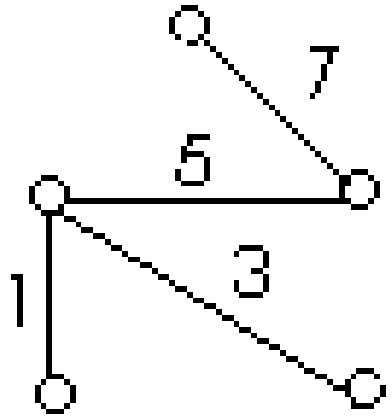
An edge e of a graph G is a cut-edge or bridge (桥) if $c(G-e)>c(G)$.

Theorem 1.2 **A connected graph has a spanning tree.**

Note. A connected graph may have many spanning trees!







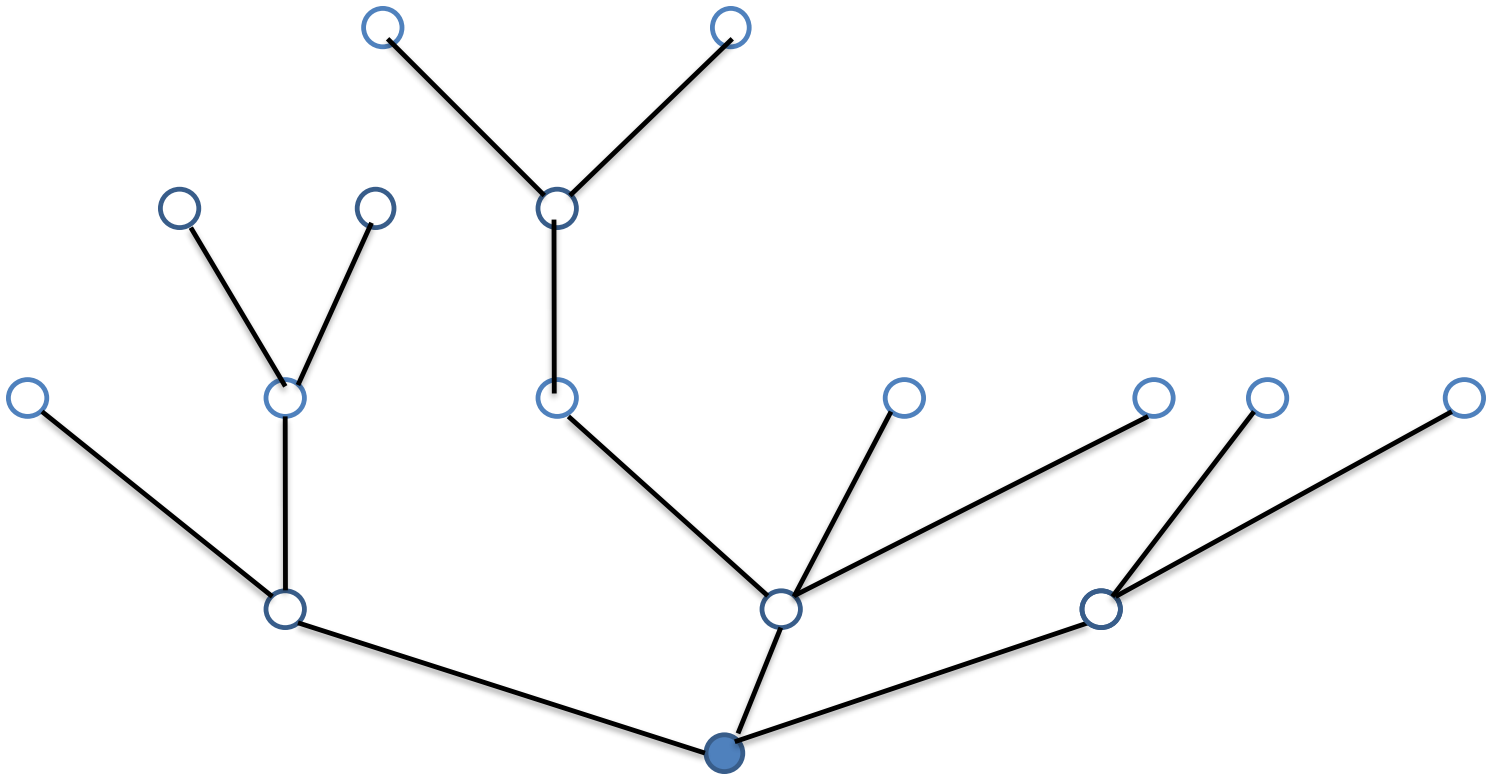
2 How to Grow Trees

Theorem 2.1 Every tree on at least two nodes has at least **two leaves**.

Tree-growing Procedure:

- Start with a single node (**root**).
- Repeat the following: If you have any graph G, create a new node (**son**) and connect it by a new edge to any node (**father of the son**) of G.

2 How to Grow Trees



Theorem 2.2 Every graph obtained by the Tree-growing Procedure is a tree, and every tree can be obtained in this way.

Theorem 2.3 Every tree on n nodes has $n-1$ edges.

例1 设$T=(V, E)$是树，如$|V|=20$，树叶共有8个，最大度数为3，问2度点和3度点各有多少?

解：设2度点为x个，则3度点数20-8-x。
因$2|E|=\sum d(v_i)$，而$|E|=|V|-1$，
故$2\times(20-1)=1\times8+2x+3(20-8-x)$，
解为2度点数x=6，3度点数为20-8-6=6。

例2 一棵最大度为4的树有两个顶点度数为2，一个顶点度数为3，三个顶点度数为4，问它有几片树叶？

解：设有x片树叶，则

顶点数$n=2+1+3+x=6+x$，

边数$m=n-1=5+x$，

在图中度数之和等于图中边数的两倍，

所以，$2(5+x)=2\cdot2+1\cdot3+3\cdot4+x\cdot1$，

解得，$x=9$。

例3 一棵树有n_2个顶点度数为2，n_3个顶点度数为3，…，n_k个顶点度数为k，问它有几片叶？

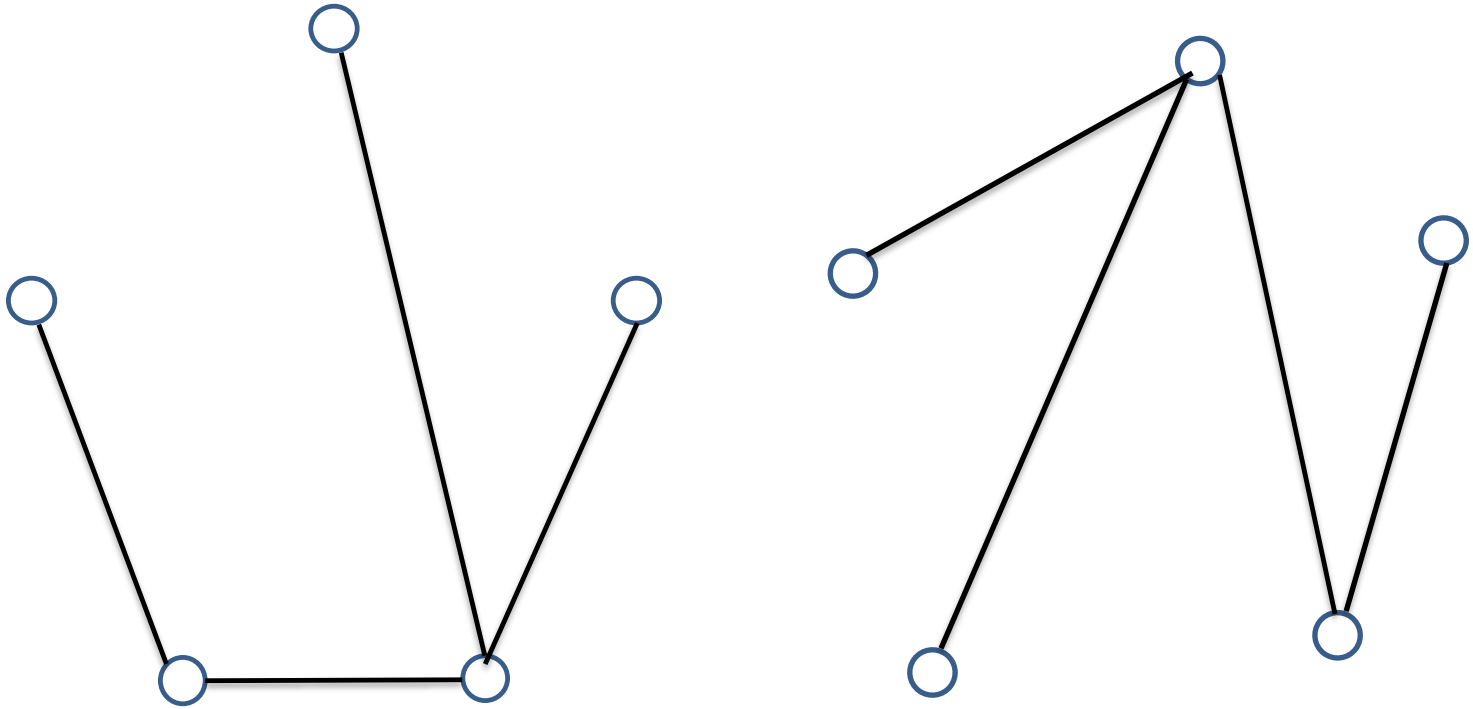
解：设它有n_1片叶，则

$$n_1+2n_2+3n_3+\cdots+kn_k=2(n_1+n_2+\ldots+n_k-1)$$

得：$n_1=n_3+2n_4+\cdots+(k-2)n_k+2$。

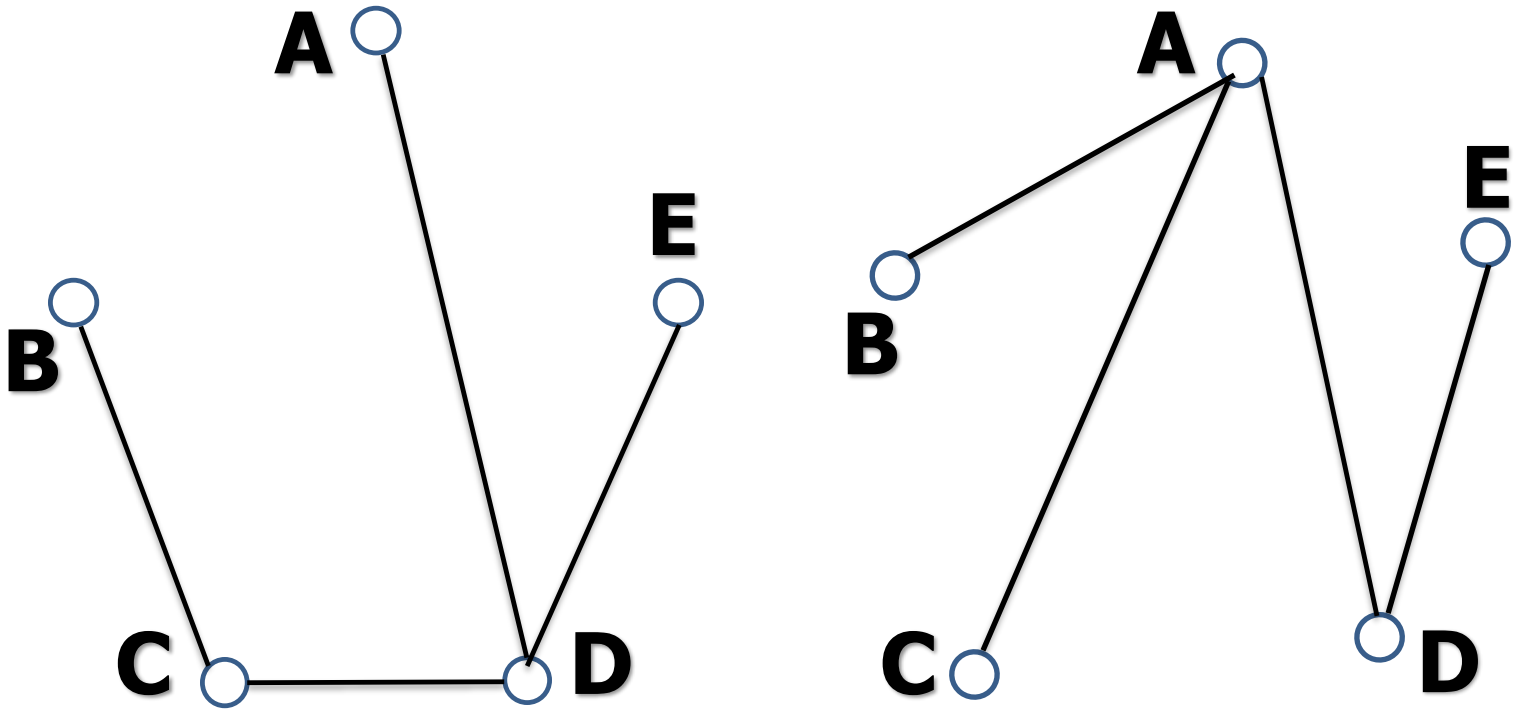
（注：图中度数之和等于边数的两倍；树中顶点数等于边数加1。）

3 How to Count Trees



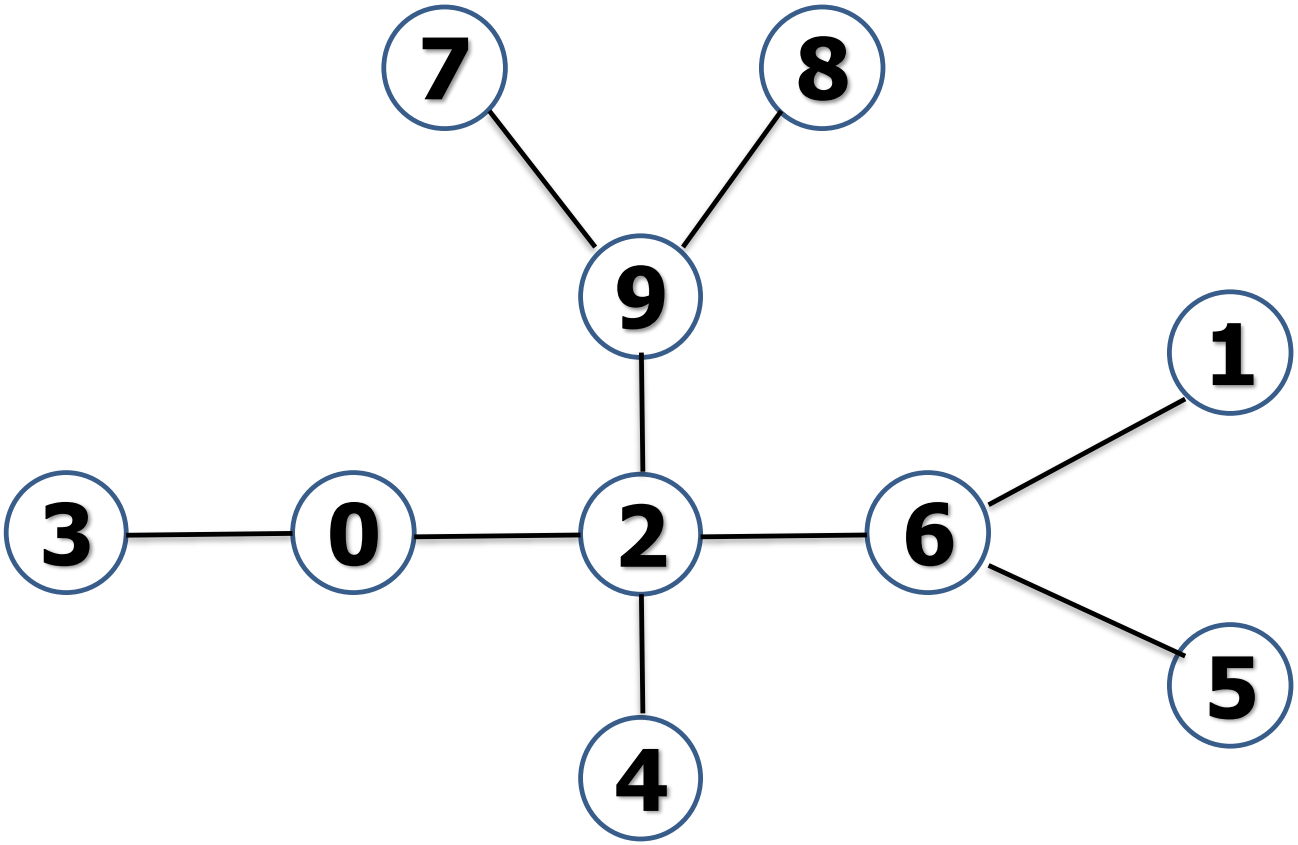
Two? unlabeled trees

3 How to Count Trees



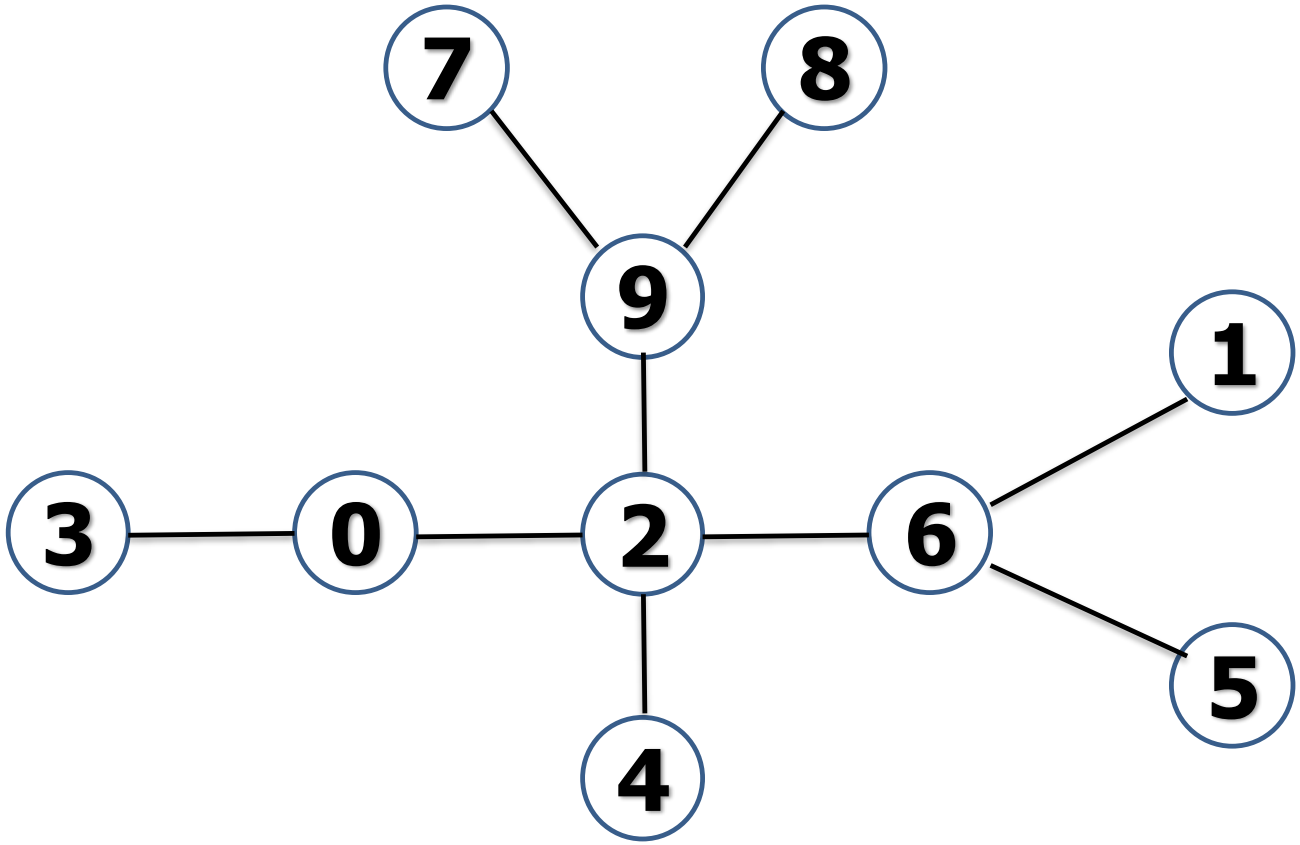
Theorem 3.1 (Cayley's Theorem)
The number of labeled trees on n nodes is n^{n-2}.

4 How to Store Trees

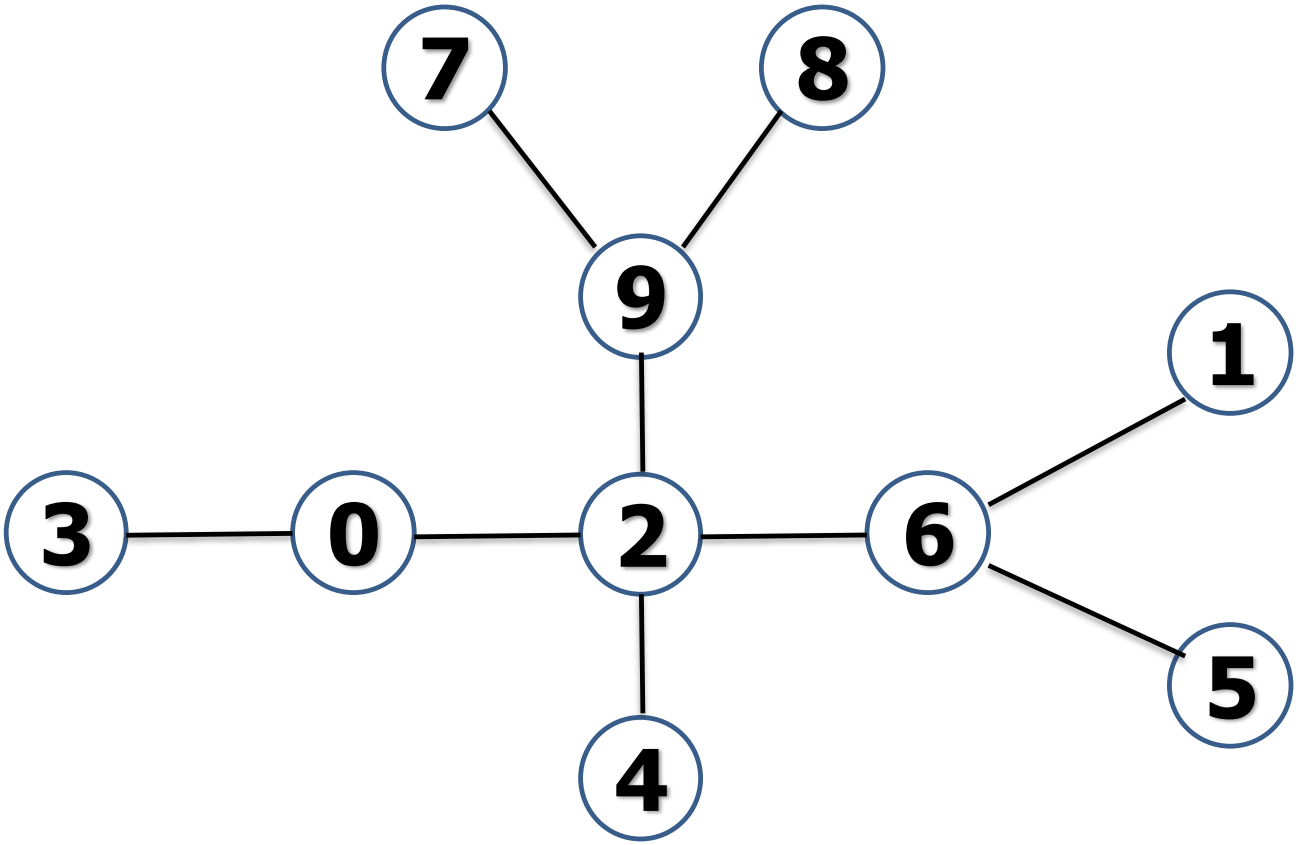


The father code:

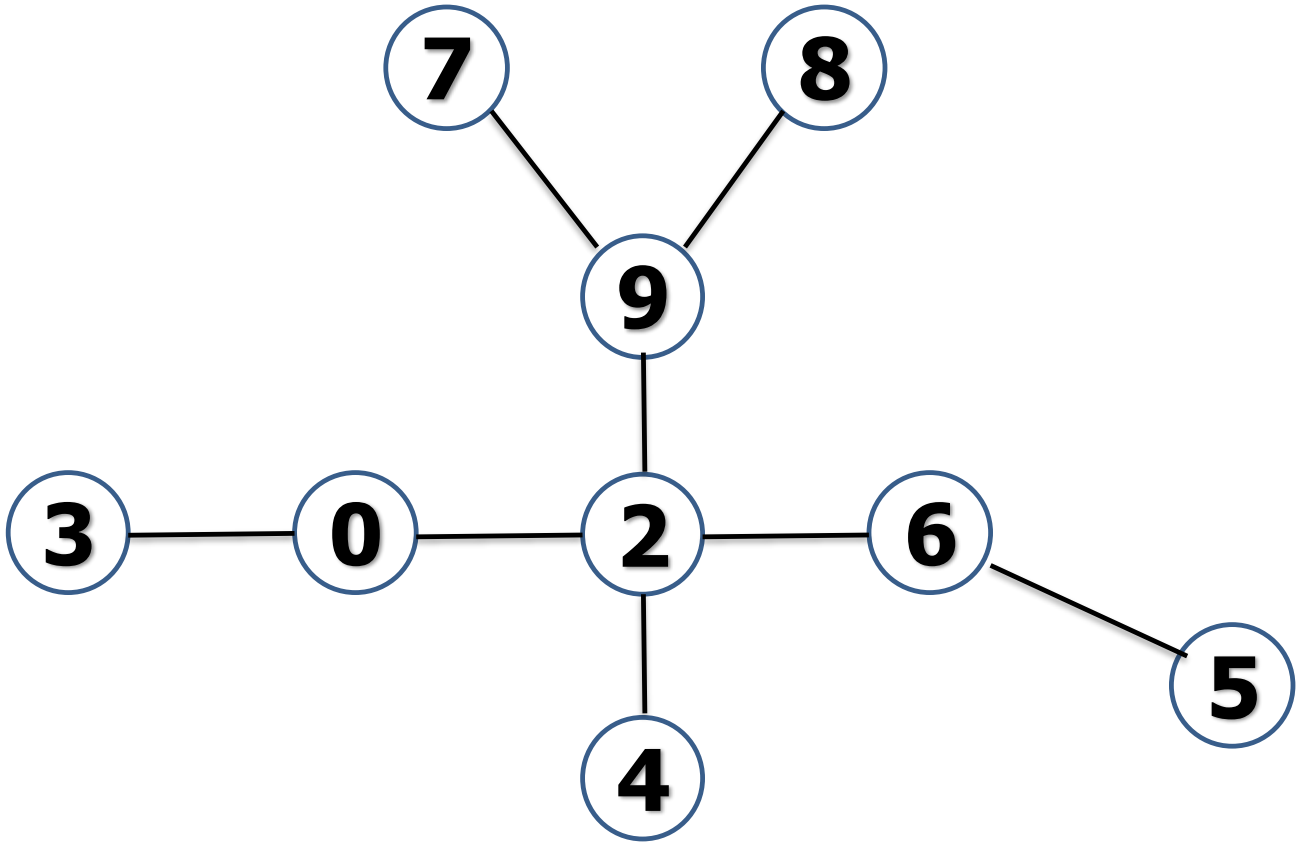
1	2	3	4	5	6	7	8	9
6	0	0	2	6	2	9	9	2



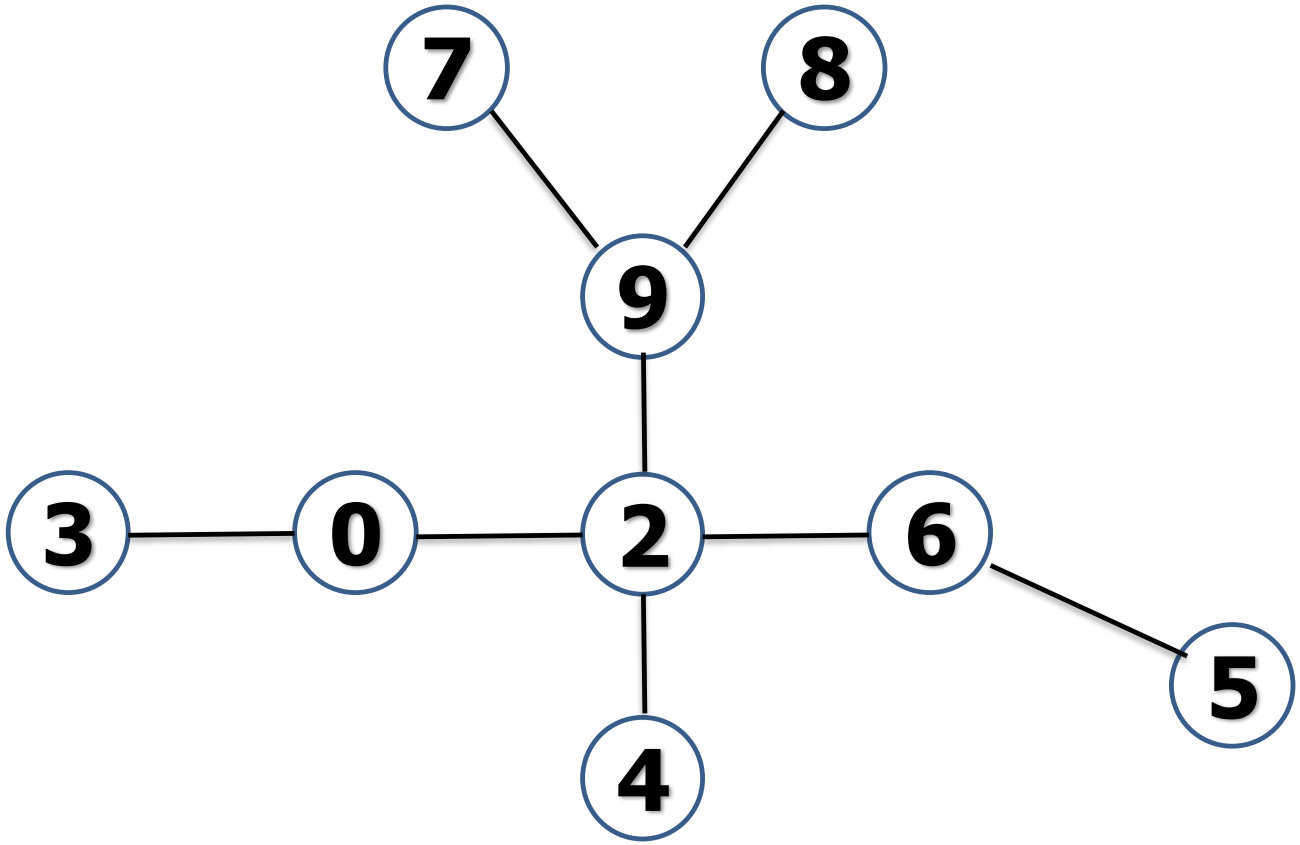
The Prüfer code: **1 3 4 5 6 7 8 9 2**
6 0 2 6 2 9 9 2 0



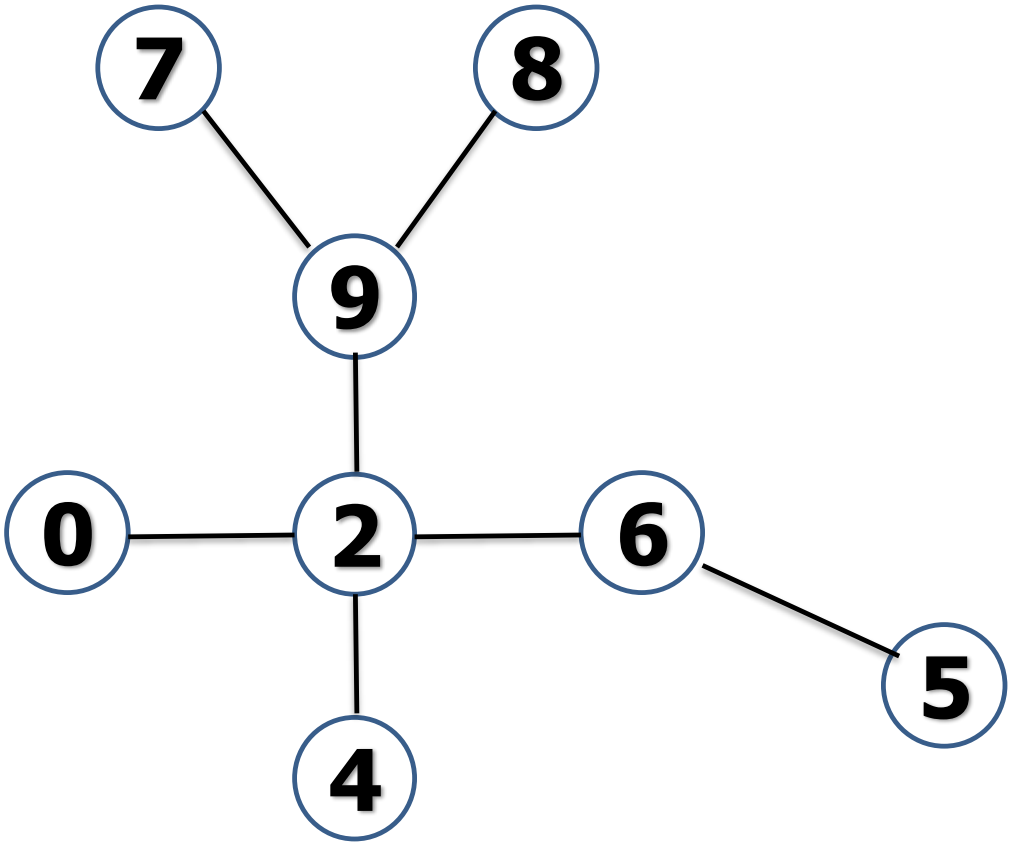
The Prüfer code: 1 6



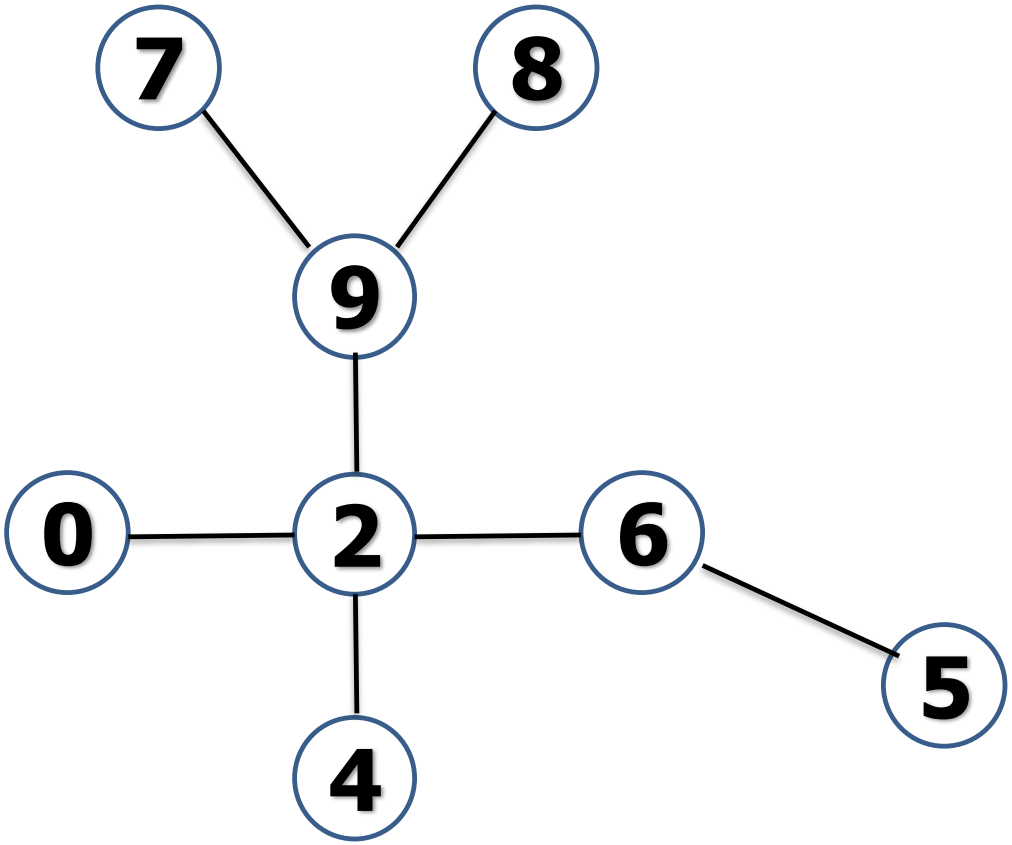
The Prüfer code: 1 6



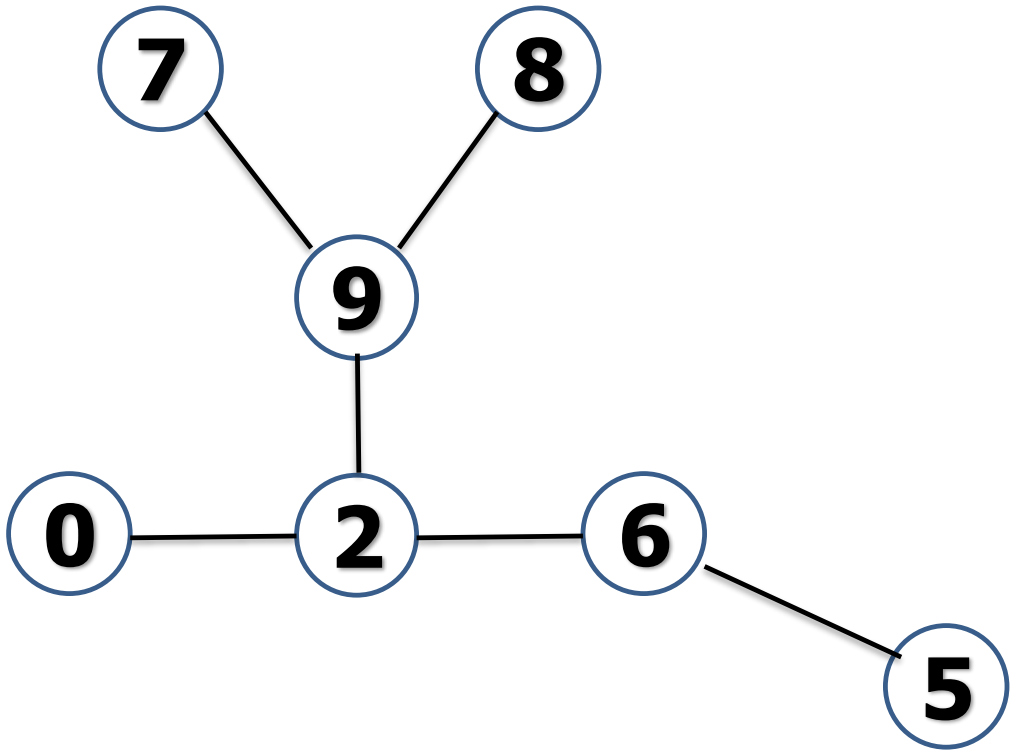
The Prüfer code: 1 3 6 0



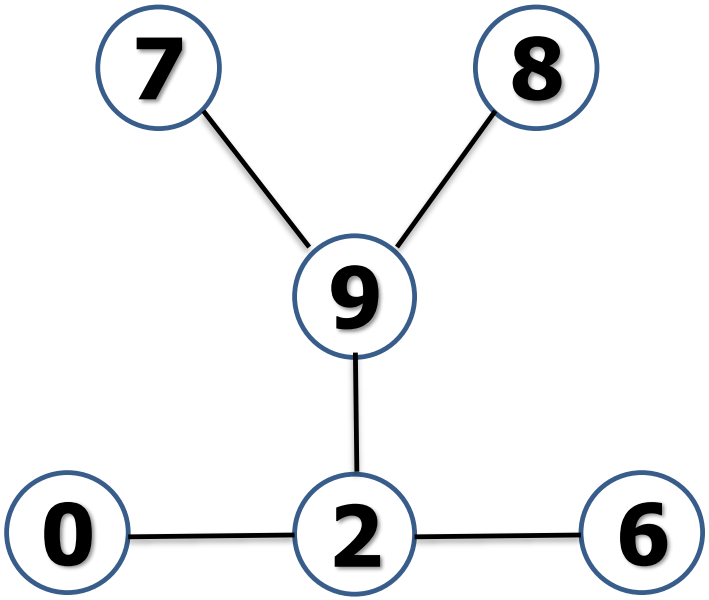
The Prüfer code: 1 3 6 0



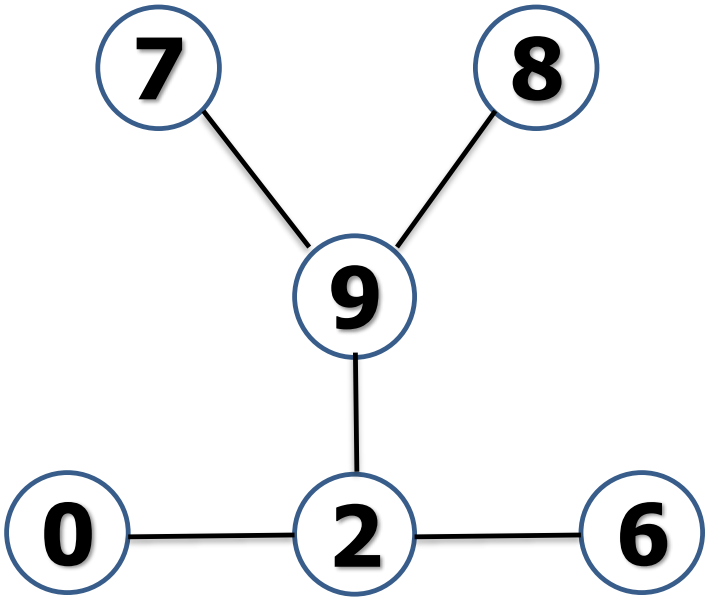
The Prüfer code: 1 3 4
6 0 2



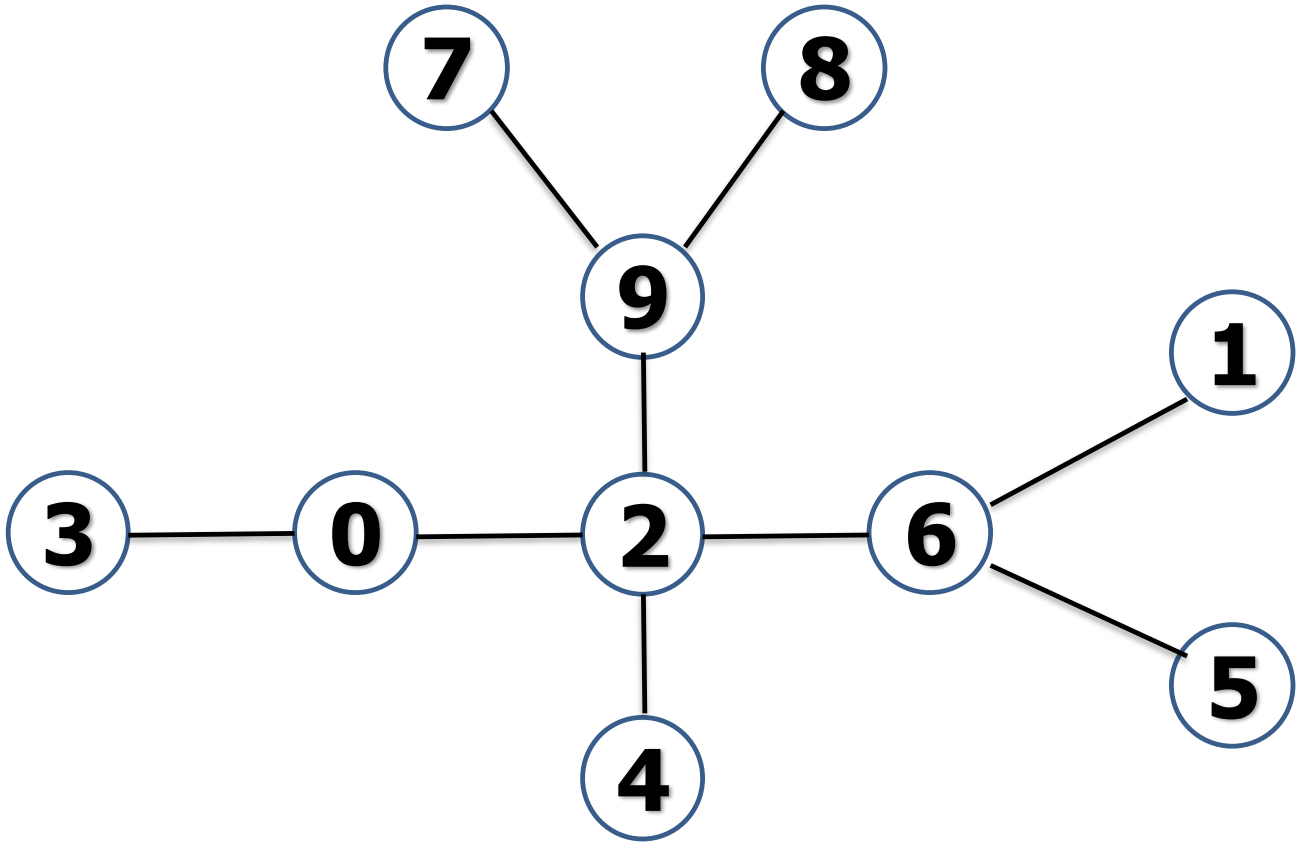
The Prüfer code: **1 3 4**
6 0 2



The Prüfer code: **1 3 4 5**
6 0 2 6



The Prüfer code: **1 3 4 5 6 7 8 9 2**
6 0 2 6 2 9 9 2 0



The Prüfer code: 1 3 4 5 6 7 8 9 2
6 0 2 6 2 9 9 2 0

Lemma 4.1 The second row of a Prüfer code determines the first.

1 6 0 2 6 2 9 9 2 0	1 3 6 0 2 6 2 9 9 2 0
1 3 4 6 0 2 6 2 9 9 2 0	1 3 4 5 6 0 2 6 2 9 9 2 0
1 3 4 5 6 6 0 2 6 2 9 9 2 0	...

The Prüfer code:

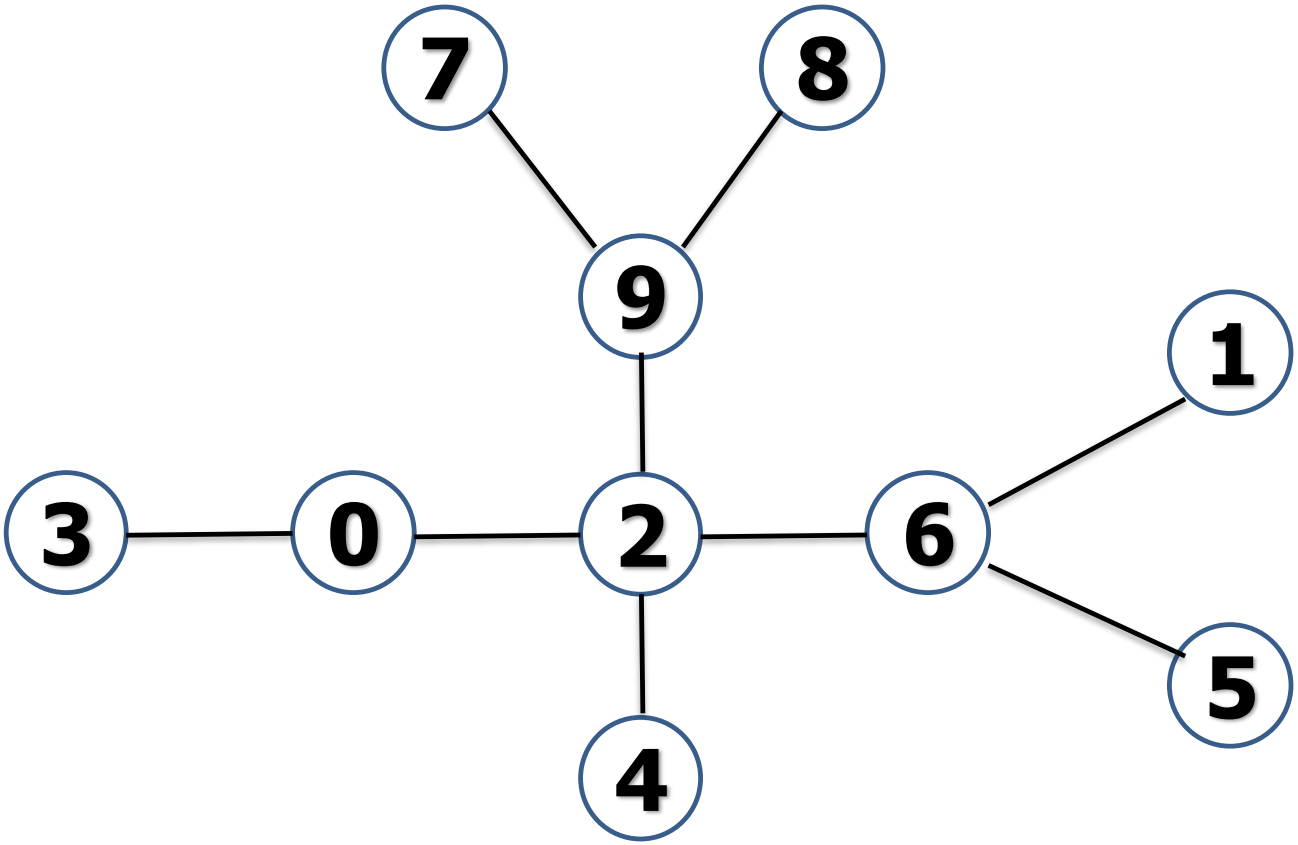
1 3 4 5 6 7 8 9 2
6 0 2 6 2 9 9 2 0

Lemma 4.1 The second row of a Prüfer code determines the first.

Proof idea.

Each entry in the first row of the Prüfer code is the smallest integer that does not occur in the first row before it, nor in the second row below or after it.

Cayley's Theorem. #labeled trees $= n^{n-2}$.



The Prüfer code: 6 0 2 6 2 9 9 2

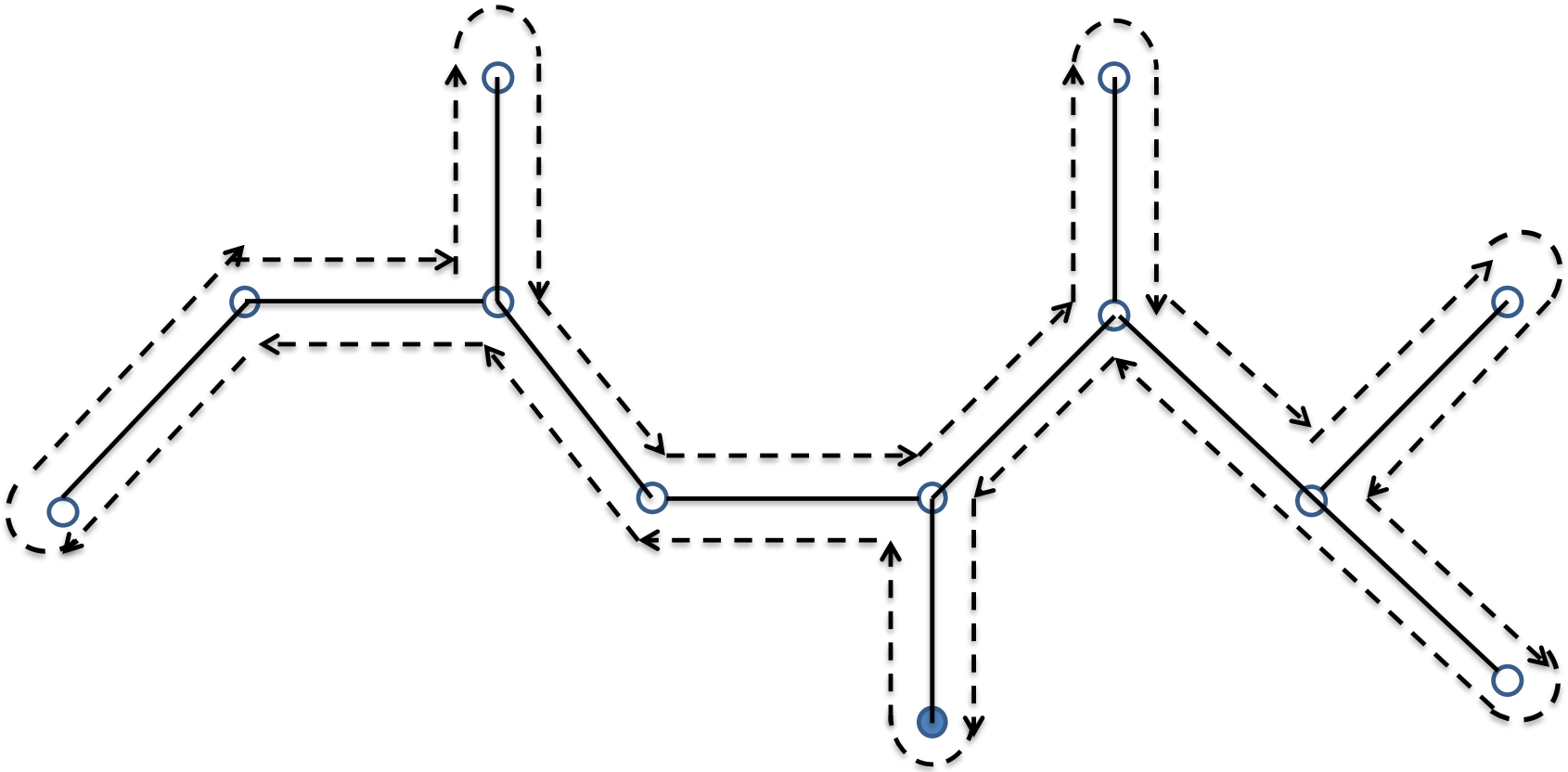
5 The Number of Unlabeled Trees

Theorem 5.1 The number T_n of unlabeled trees with n nodes satisfies

$$n^{n-2}/n! \le T_n \le C(2n-4,n-2).$$

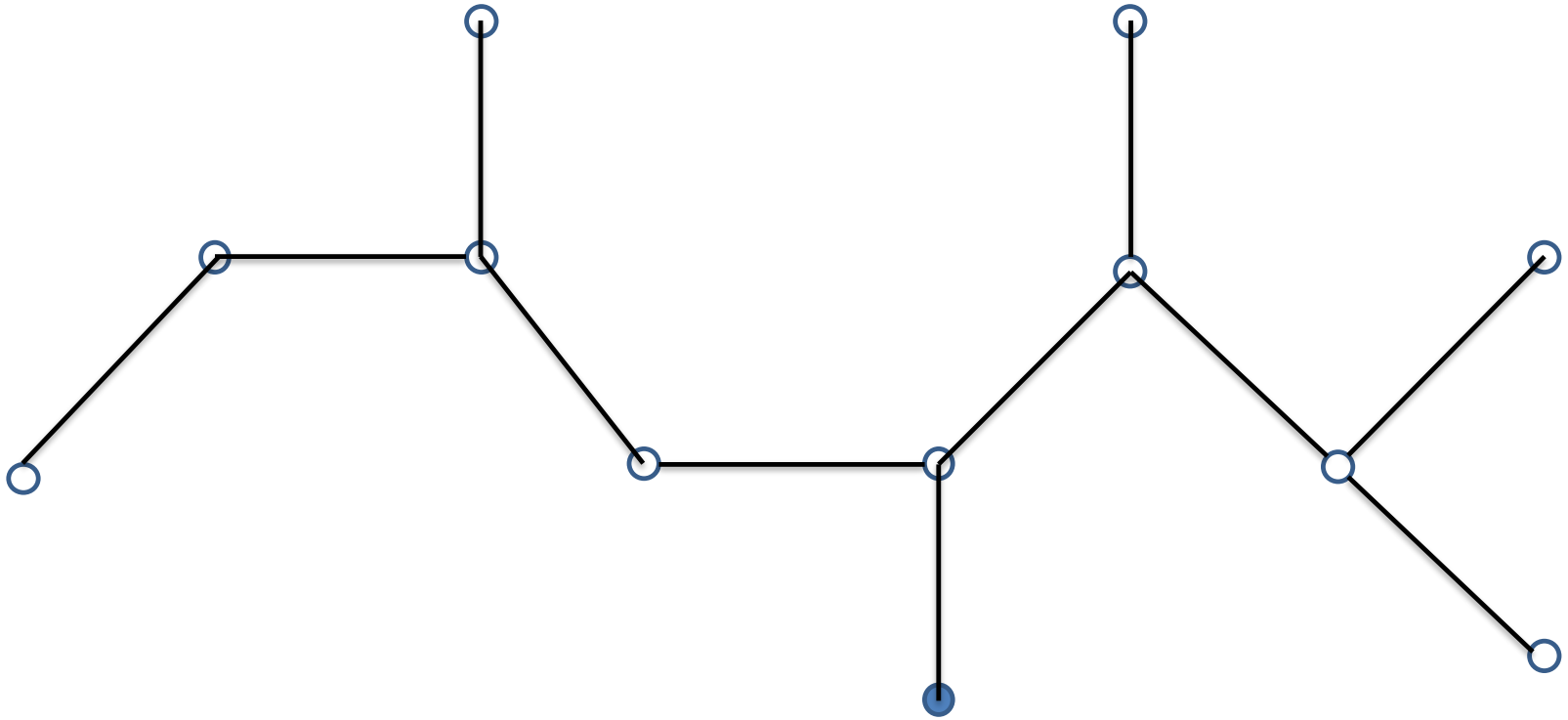
(George Pólya: $T_n \sim ab^n/n^{5/2}$, where a=0.5349… and b=2.9557…)

Proof. The lower bound is easy, since the nodes of an unlabeled tree can be labeled in $n!$ ways.



The planar code:

1111100100011011010000



The planar code:

1111100100011011010000

Erdös-Sós Conjecture (1962)

Every simple graph with average degree greater than k-1 contains every tree with k edges.