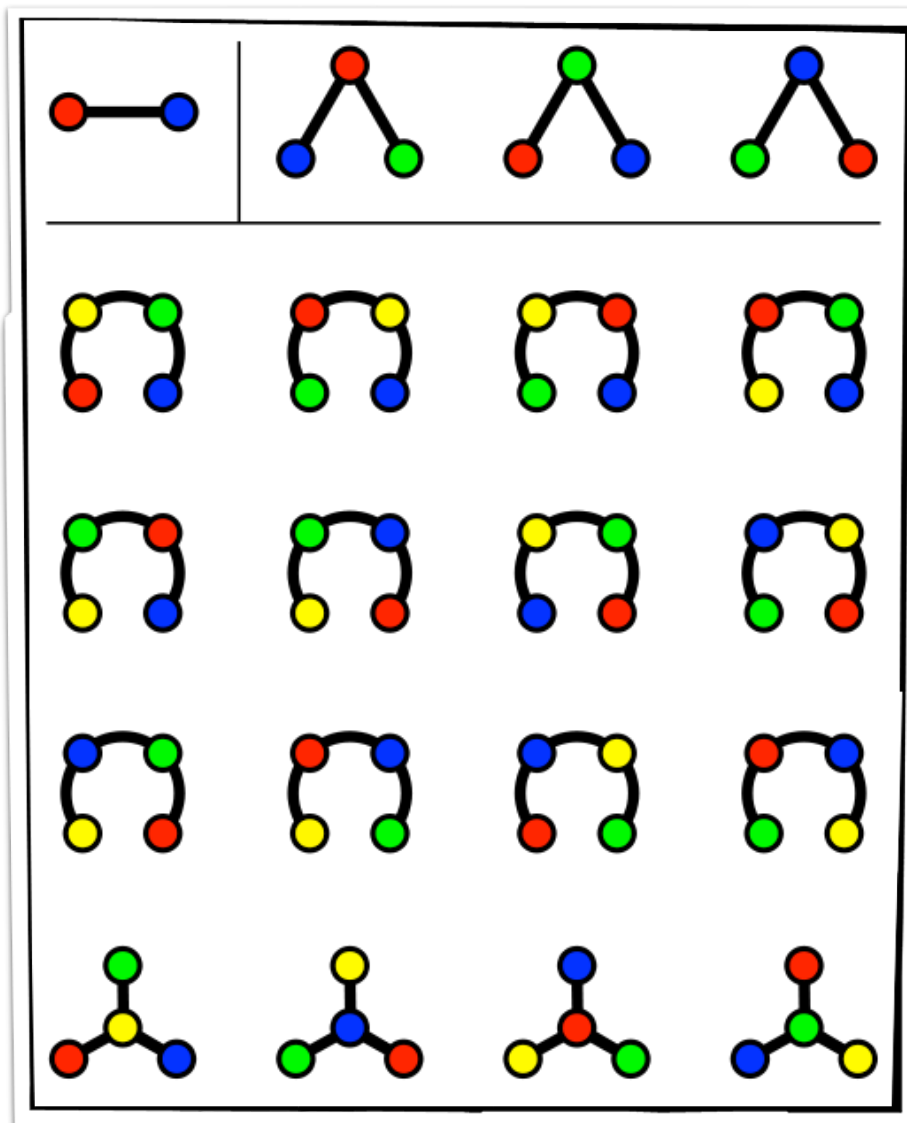


Combinatorics

Cayley Formula

尹一通 Nanjing University, 2025 Spring

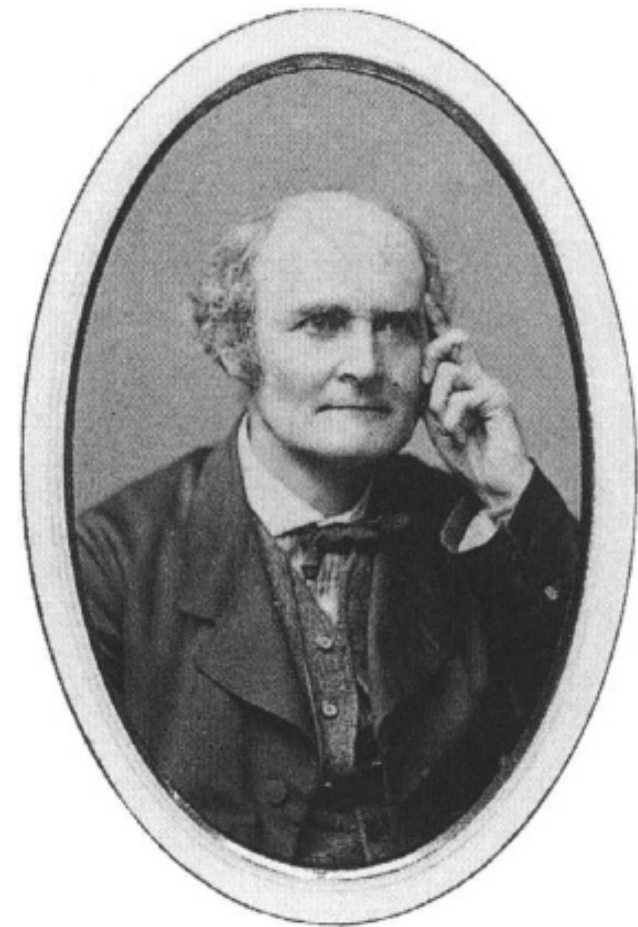
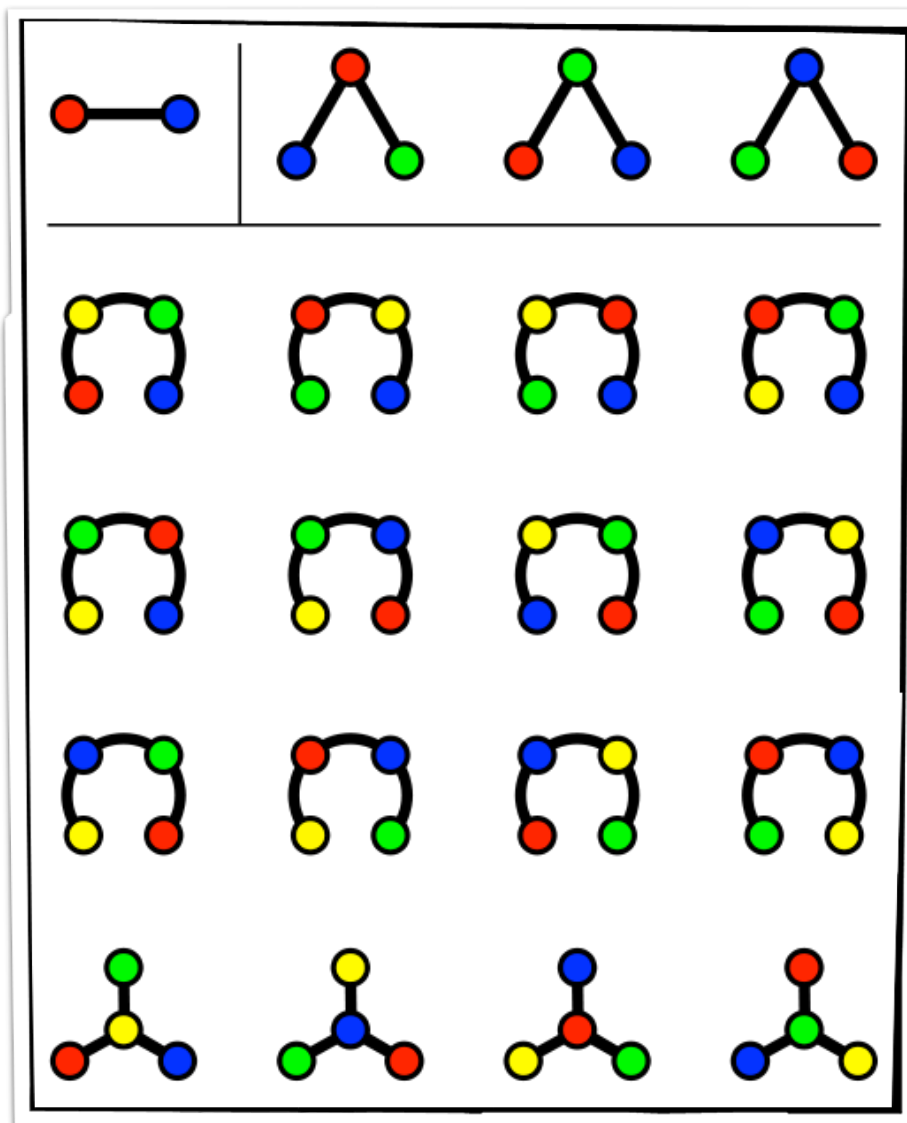
Counting (labeled) trees



*“How many different trees
can be formed from
 n **distinct** vertices?”*

Cayley's formula:

There are n^{n-2} trees on n distinct vertices.



Arthur Cayley

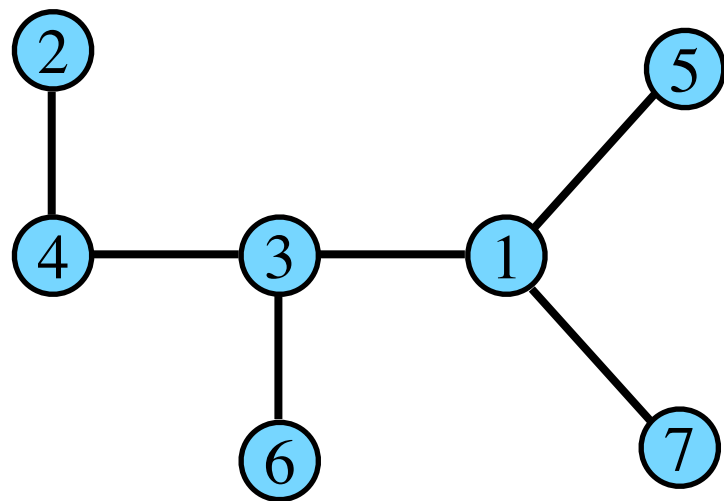
Cayley Formula: **Prüfer Code**

Prüfer Code

leaf : vertex of degree 1

removing a leaf from T still gives a tree

$T_1 :$



$u_i :$ 2

$v_i :$ 4

$T_1 = T ;$

for $i = 1$ to $n-1$

$u_i :$ smallest leaf in $T_i ;$

$(u_i, v_i) :$ edge in $T_i ;$

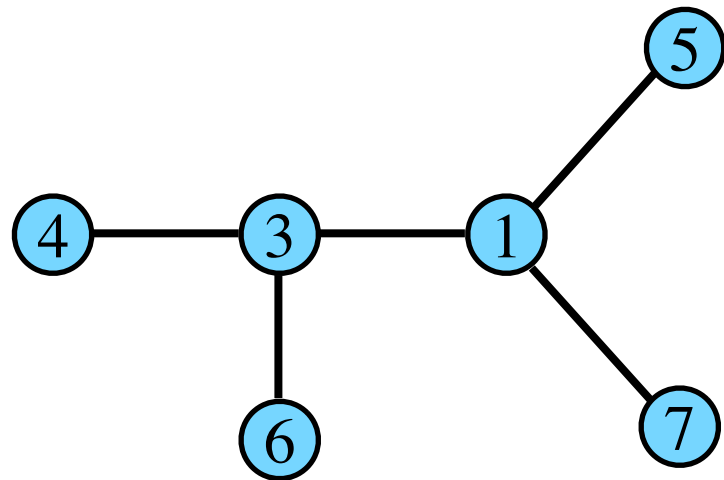
$T_{i+1} = \text{delete } u_i \text{ from } T_i ;$

Prüfer Code

leaf : vertex of degree 1

removing a leaf from T still gives a tree

T_2 :



u_i : 2, 4

v_i : 4, 3

$T_1 = T$;

for $i = 1$ to $n-1$

u_i : smallest leaf in T_i ;

(u_i, v_i) : edge in T_i ;

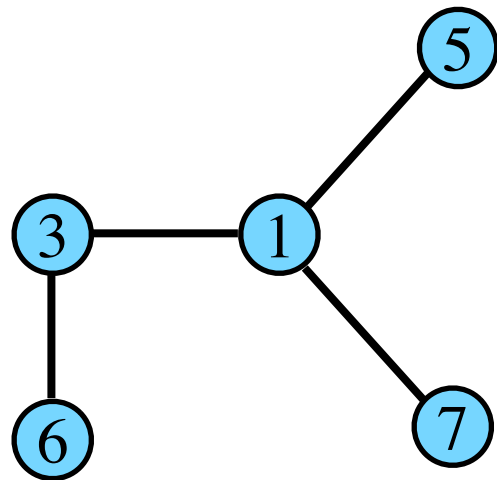
$T_{i+1} = \text{delete } u_i \text{ from } T_i$;

Prüfer Code

leaf : vertex of degree 1

removing a leaf from T still gives a tree

T_3 :



u_i : 2, 4, 5

v_i : 4, 3, 1

$T_1 = T$;

for $i = 1$ to $n-1$

u_i : smallest leaf in T_i ;

(u_i, v_i) : edge in T_i ;

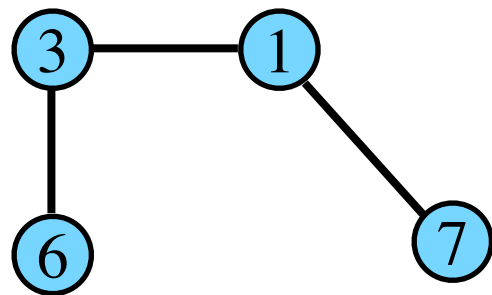
$T_{i+1} = \text{delete } u_i \text{ from } T_i$;

Prüfer Code

leaf : vertex of degree 1

removing a leaf from T still gives a tree

T_4 :



u_i : 2, 4, 5, 6

v_i : 4, 3, 1, 3

$T_1 = T$;

for $i = 1$ to $n-1$

u_i : smallest leaf in T_i ;

(u_i, v_i) : edge in T_i ;

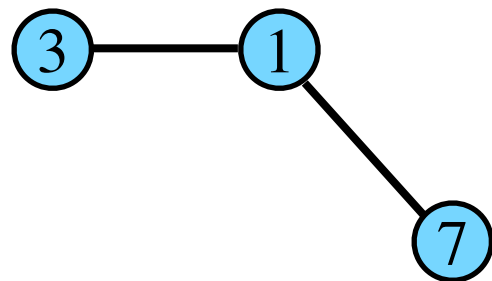
$T_{i+1} = \text{delete } u_i \text{ from } T_i$;

Prüfer Code

leaf : vertex of degree 1

removing a leaf from T still gives a tree

T_5 :



u_i : 2, 4, 5, 6, 3

v_i : 4, 3, 1, 3, 1

$T_1 = T$;

for $i = 1$ to $n-1$

u_i : smallest leaf in T_i ;

(u_i, v_i) : edge in T_i ;

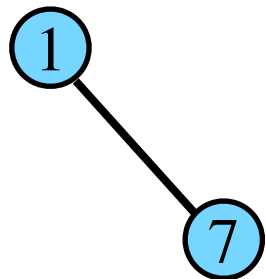
$T_{i+1} = \text{delete } u_i \text{ from } T_i$;

Prüfer Code

leaf : vertex of degree 1

removing a leaf from T still gives a tree

T_6 :



u_i : 2, 4, 5, 6, 3, 1

v_i : 4, 3, 1, 3, 1, 7

$T_1 = T$;

for $i = 1$ to $n-1$

u_i : smallest leaf in T_i ;

(u_i, v_i) : edge in T_i ;

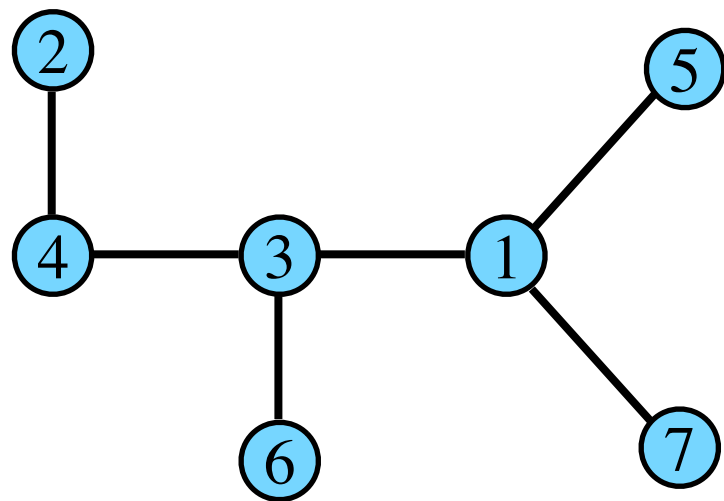
$T_{i+1} = \text{delete } u_i \text{ from } T_i$;

Prüfer Code

leaf : vertex of degree 1

removing a leaf from T still gives a tree

T :



u_i : 2, 4, 5, 6, 3, 1

v_i : 4, 3, 1, 3, 1, 7

$T_1 = T$;

for $i = 1$ to $n-1$

u_i : smallest leaf in T_i ;

(u_i, v_i) : edge in T_i ;

$T_{i+1} = \text{delete } u_i \text{ from } T_i$;

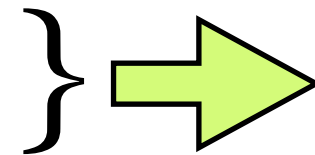
Prüfer code:

$(v_1, v_2, \dots, v_{n-2})$

edges of $T : (u_i, v_i), 1 \leq i \leq n-1$

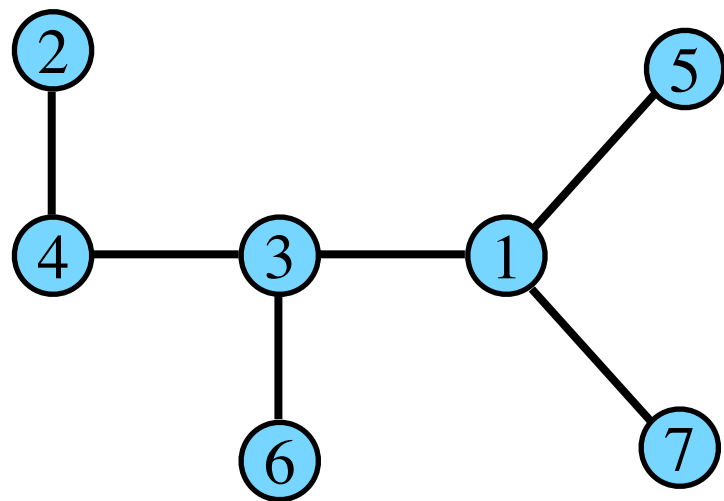
$$v_{n-1} = n$$

u_i : smallest leaf in T_i
a tree has ≥ 2 leaves



n is never deleted
 $u_i \neq n$

$T :$



$u_i : 2, 4, 5, 6, 3, 1$

$v_i : 4, 3, 1, 3, 1, 7$

$(v_1, v_2, \dots, v_{n-2})$

Only need to recover
every u_i from $(v_1, v_1, \dots, v_{n-2})$.

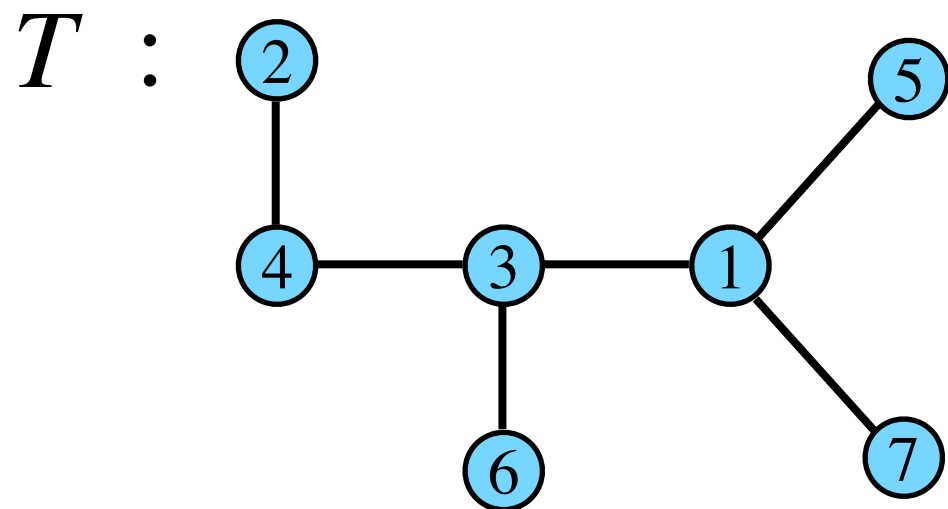
u_i is the smallest number *not* in
 $\{u_1, \dots, u_{i-1}\} \cup \{v_i, \dots, v_{n-1}\}$

u_i is the smallest number *not* in
 $\{u_1, \dots, u_{i-1}\} \cup \{v_i, \dots, v_{n-1}\}$

\forall vertex v in T ,

occurrences of v in $u_1, u_2, \dots, u_{n-1}, v_{n-1}$: **1**

occurrences of v in edges (u_i, v_i) , $1 \leq i \leq n-1$: **$\deg_T(v)$**



occurrences of v in
Prüfer code: $(v_1, v_2, \dots, v_{n-2})$

$\deg_T(v)-1$

u_i : 2, 4, 5, 6, 3, 1

v_i : 4, 3, 1, 3, 1, 7

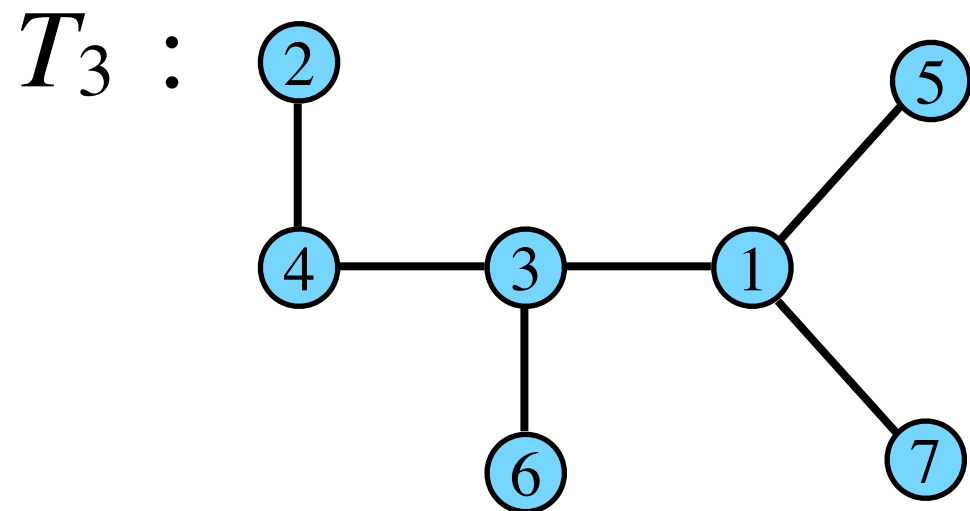
$(v_1, v_2, \dots, v_{n-2})$

u_i is the smallest number *not* in
 $\{u_1, \dots, u_{i-1}\} \cup \{v_i, \dots, v_{n-1}\}$

\forall vertex v in T ,

occurrences of v in $u_1, u_2, \dots, u_{n-1}, v_{n-1}$: **1**

occurrences of v in edges (u_i, v_i) , $1 \leq i \leq n-1$: **$\deg_T(v)$**



occurrences of v in (v_i, \dots, v_{n-2})

$\deg_{T_i}(v) - 1$

leaf v of T_i :

in $\{u_i, u_{i+1}, \dots, u_{n-1}, v_{n-1}\}$

not in $\{v_i, v_{i+1}, \dots, v_{n-2}\}$

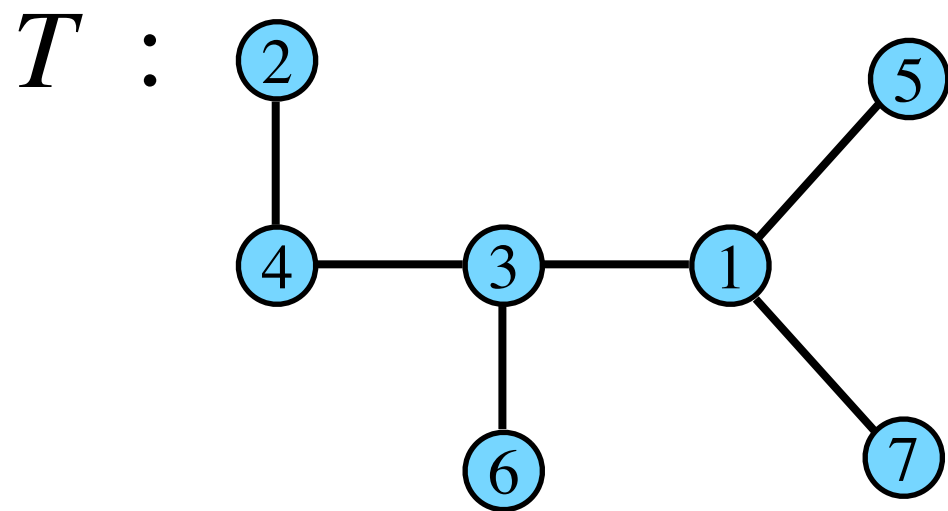
u_i : 2, 4, 5, 6, 3, 1

v_i : 4, 3, 1, 3, 1, 7

$(v_1, v_2, \dots, v_{n-2})$

u_i : smallest leaf in T_i

u_i is the smallest number *not* in
 $\{u_1, \dots, u_{i-1}\} \cup \{v_i, \dots, v_{n-1}\}$



u_i : 2, 4, 5, 6, 3, 1

v_i : 4, 3, 1, 3, 1, 7

$(v_1, v_2, \dots, v_{n-2})$

$T = \text{empty graph};$

$v_{n-1} = n$;

for $i = 1$ to $n-1$

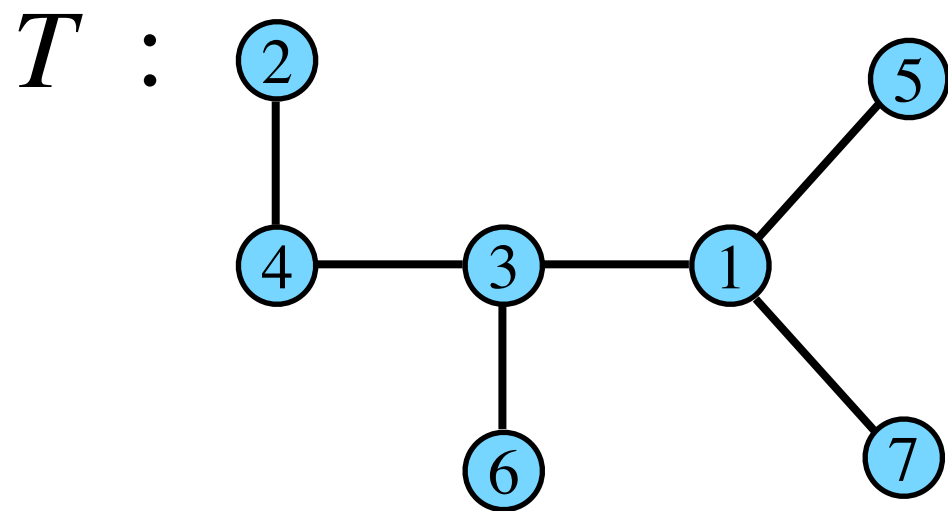
u_i : smallest number not in
 $\{u_1, \dots, u_{i-1}\} \cup \{v_i, \dots, v_{n-1}\}$

add edge (u_i, v_i) to T ;

Prüfer code is reversible \Rightarrow 1-1

every $(v_1, v_2, \dots, v_{n-2}) \in \{1, 2, \dots, n\}^{n-2}$

is decodable to a tree \Rightarrow onto



u_i : 2, 4, 5, 6, 3, 1

v_i : 4, 3, 1, 3, 1, 7

$(v_1, v_2, \dots, v_{n-2})$

$T =$ empty graph;

$v_{n-1} = n$;

for $i = 1$ to $n-1$

u_i : smallest number not in
 $\{u_1, \dots, u_{i-1}\} \cup \{v_i, \dots, v_{n-1}\}$

add edge (u_i, v_i) to T ;

Prüfer code is reversible  1-1

every $(v_1, v_2, \dots, v_{n-2}) \in \{1, 2, \dots, n\}^{n-2}$

is decodable to a tree  onto

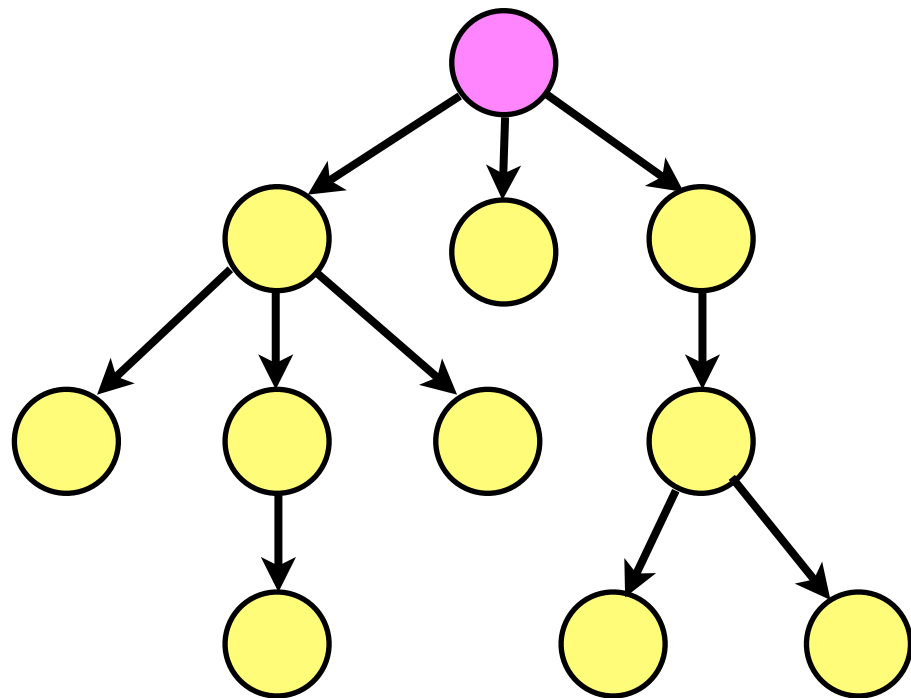
Cayley's formula:

There are n^{n-2} trees on n distinct vertices.

Cayley Formula: **Double Counting**

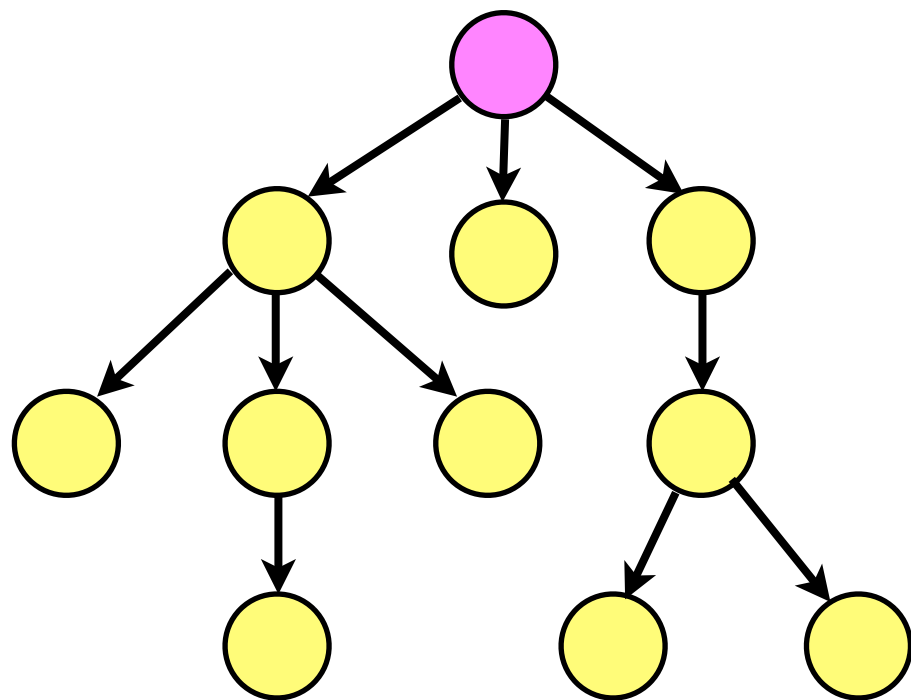
Double Counting

of *sequences* of adding *directed edges* to an empty graph to form a *rooted tree*



T_n : # of trees on n distinct vertices.

of *sequences* of adding *directed edges* to an empty graph to form a *rooted tree*



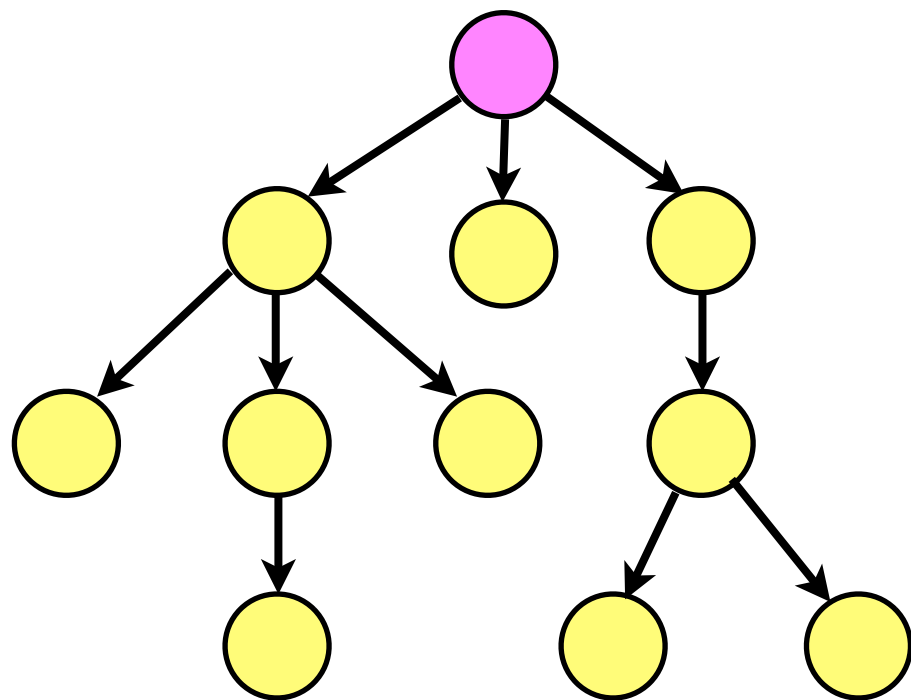
From a tree:

- pick a root;
- pick an order of edges.

$$T_n n(n-1)!$$
$$= n! T_n$$

T_n : # of trees on n distinct vertices.

of *sequences* of adding *directed edges* to an empty graph to form a *rooted tree*

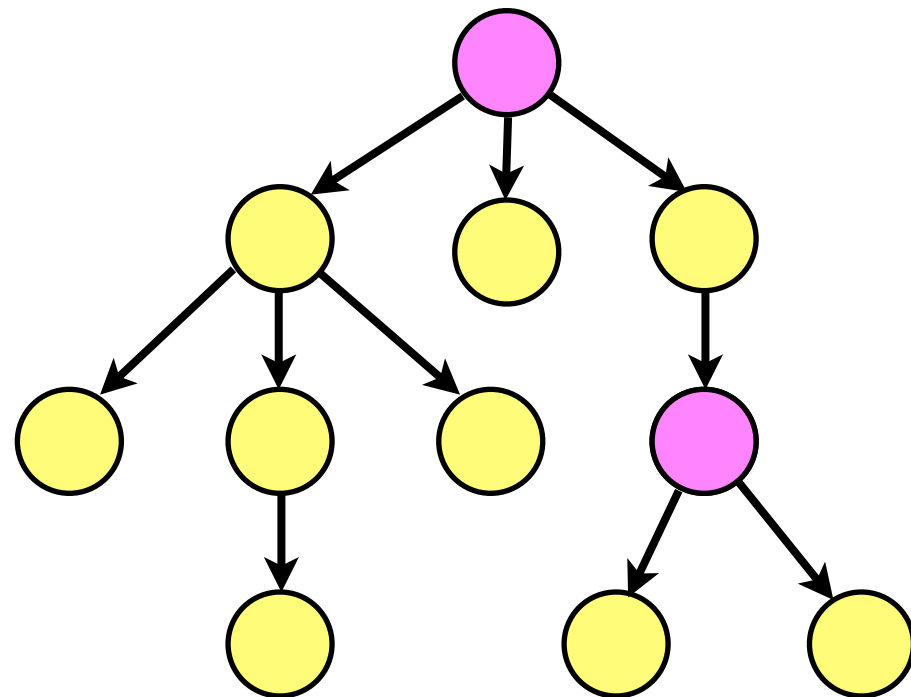


From an empty graph:

- add edges one by one

of *sequences* of adding *directed edges* to an empty graph to form a *rooted tree*

From an empty graph: • add edges one by one



Start from n ~~isolated vertices~~
rooted trees

Each step joins 2 trees.

of *sequences* of adding *directed edges* to an empty graph to form a *rooted tree*

From an empty graph: • add edges one by one

Start from n rooted trees.

After adding k edges

$n-k$ rooted trees

add an edge

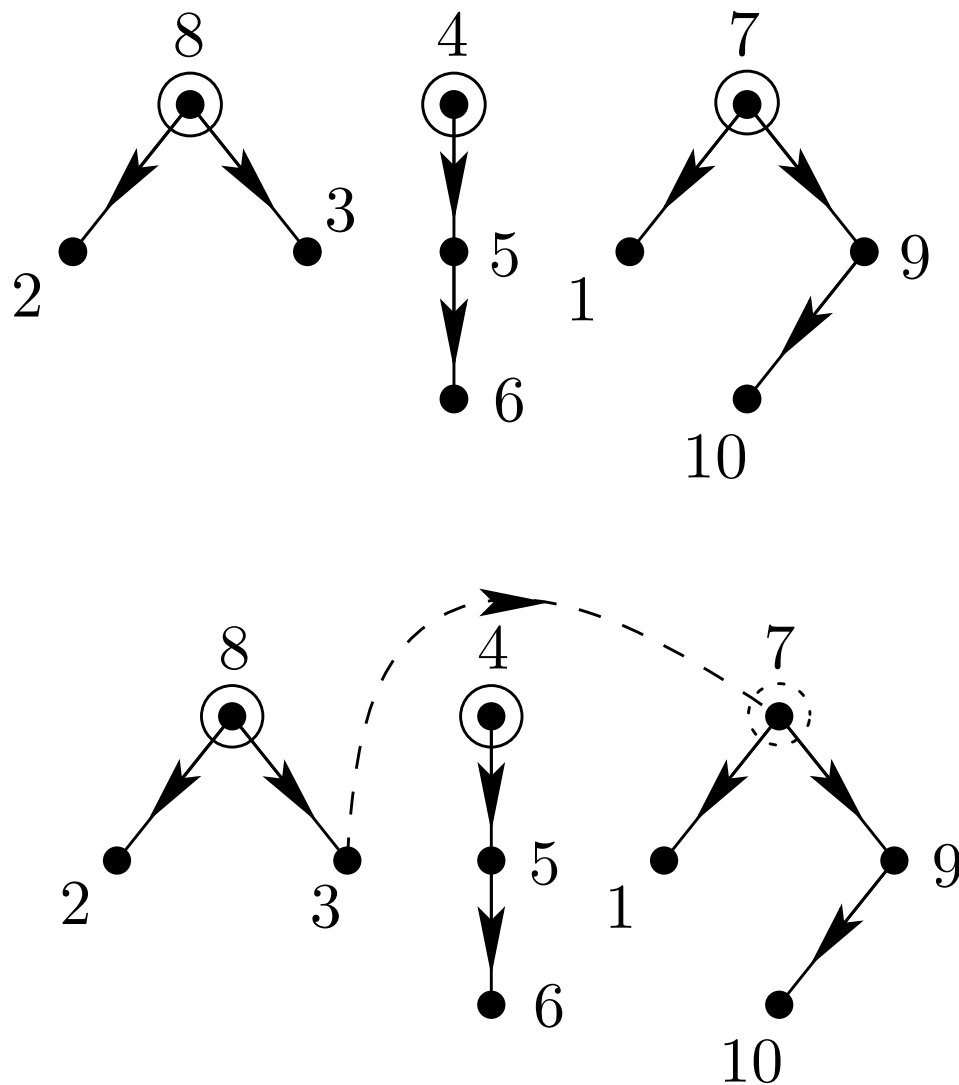
any
vertex



root of
another tree

n

$n-k-1$



of *sequences* of adding *directed edges* to an empty graph to form a *rooted tree*

From an empty graph: • add edges one by one

$$\prod_{k=0}^{n-2} n(n-k-1)$$

Start from n rooted trees.

After adding k edges

$n-k$ rooted trees

add an edge

any
vertex



root of
another tree

n

$n-k-1$

$$= n^{n-1} \prod_{k=1}^{n-1} k$$

$$= n^{n-2} n!$$

of *sequences* of adding *directed edges* to an empty graph to form a *rooted tree*

From a tree:

- pick a root;
- pick an order of edges.

$$T_n n(n-1)!$$
$$= n! T_n$$

=

From an empty graph:

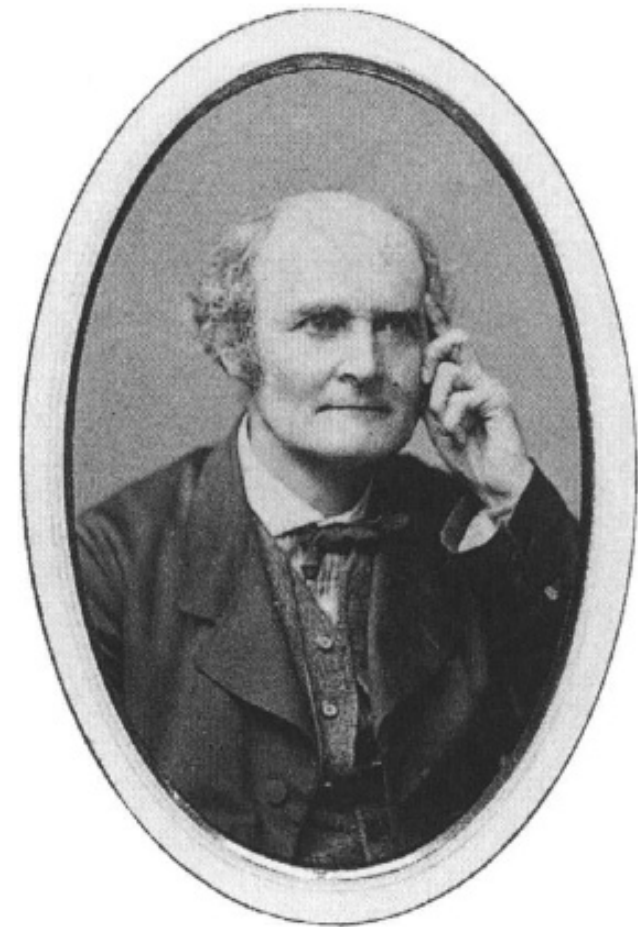
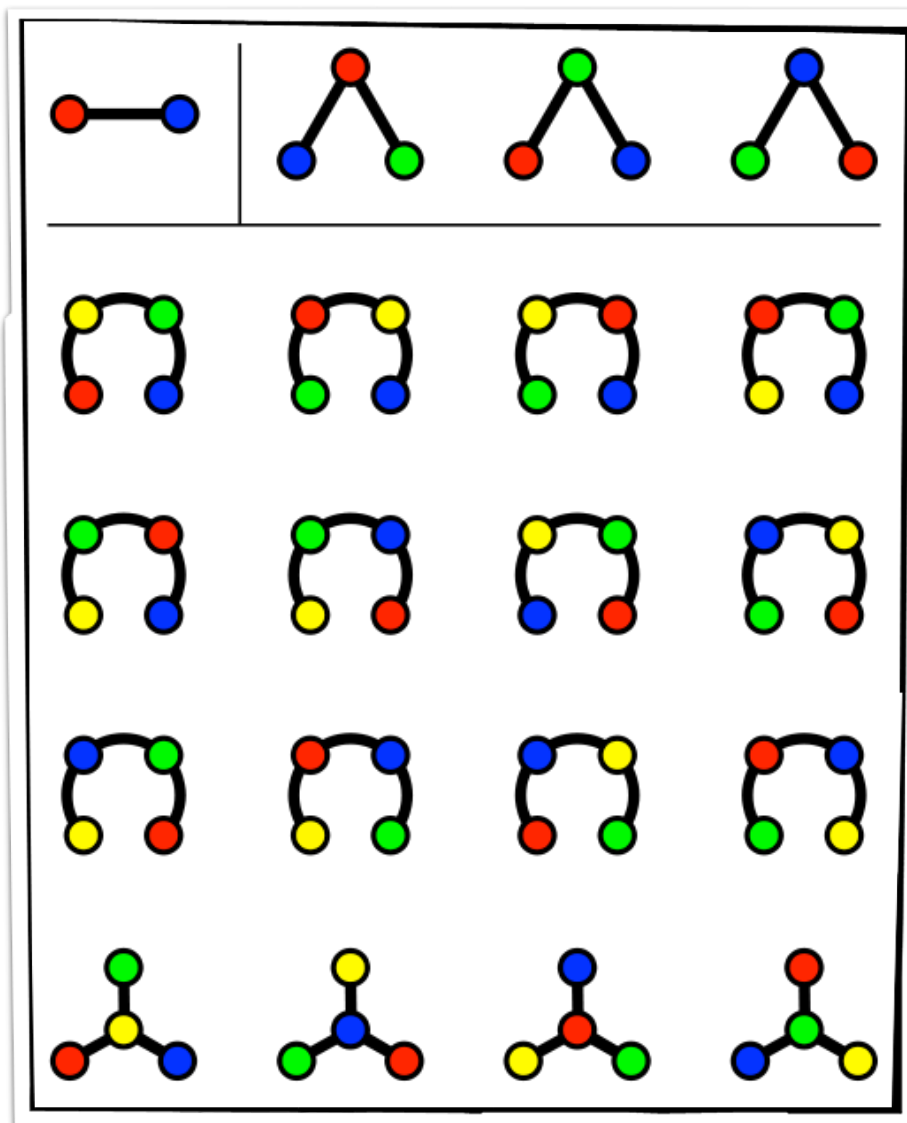
- add edges one by one

$$\prod_{k=0}^{n-2} n(n-k-1)$$
$$= n^{n-2} n!$$

$$T_n = n^{n-2}$$

Cayley's formula:

There are n^{n-2} trees on n distinct vertices.



Arthur Cayley

Cayley Formula:

Matrix-Tree Theorem

Graph *Laplacian*

Graph $G(V,E)$

adjacency matrix A

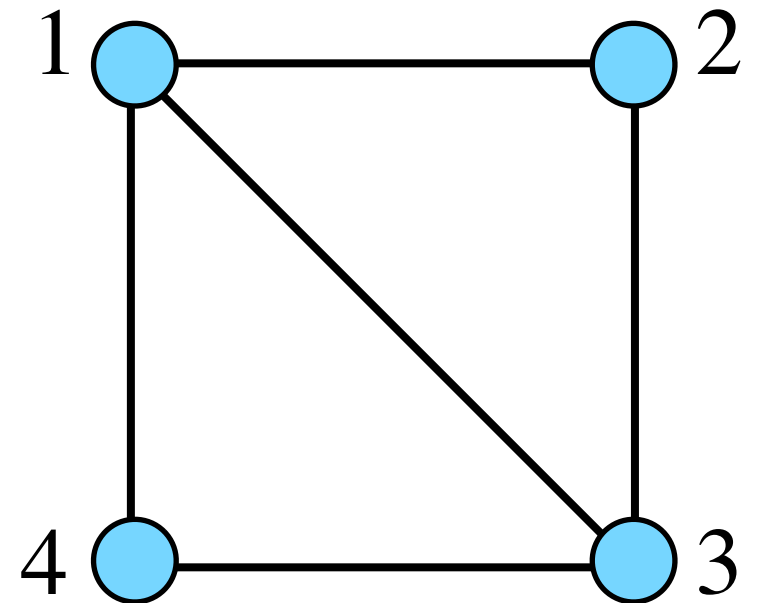
$$A(i, j) = \begin{cases} 1 & \{i, j\} \in E \\ 0 & \{i, j\} \notin E \end{cases}$$

diagonal matrix D

$$D(i, j) = \begin{cases} \deg(i) & i = j \\ 0 & i \neq j \end{cases}$$

graph *Laplacian* L

$$L = D - A$$



$$D = \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & 0 & \\ & 0 & \ddots & \\ & & & d_n \end{bmatrix}$$

$$L = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix}$$

Graph *Laplacian*

graph *Laplacian* L

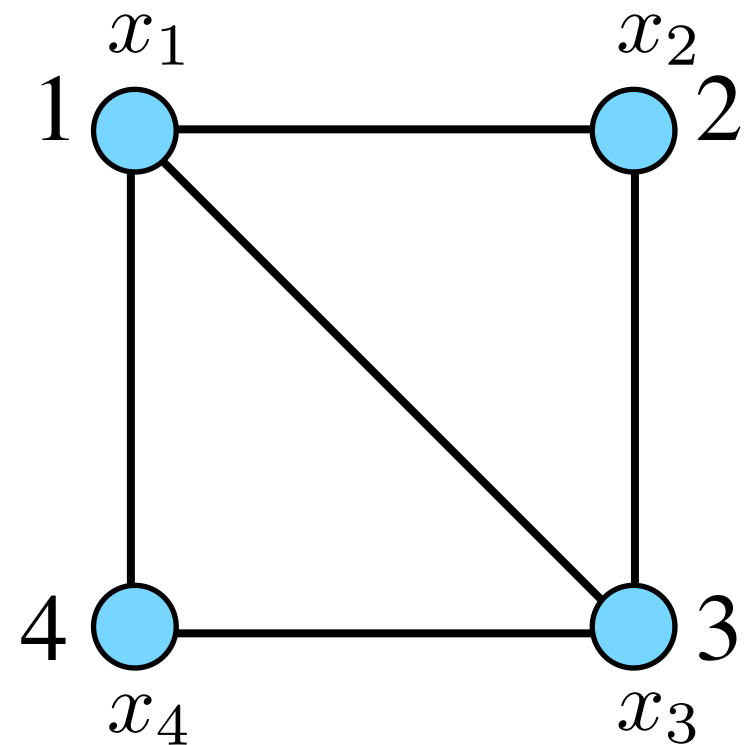
$$L(i, j) = \begin{cases} \deg(i) & i = j \\ -1 & i \neq j, \{i, j\} \in E \\ 0 & \text{otherwise} \end{cases}$$

quadratic form:

$$xLx^T = \sum_i d_i x_i^2 - \sum_{ij \in E} x_i x_j = \frac{1}{2} \sum_{ij \in E} (x_i - x_j)^2$$

incidence matrix $B : n \times m$

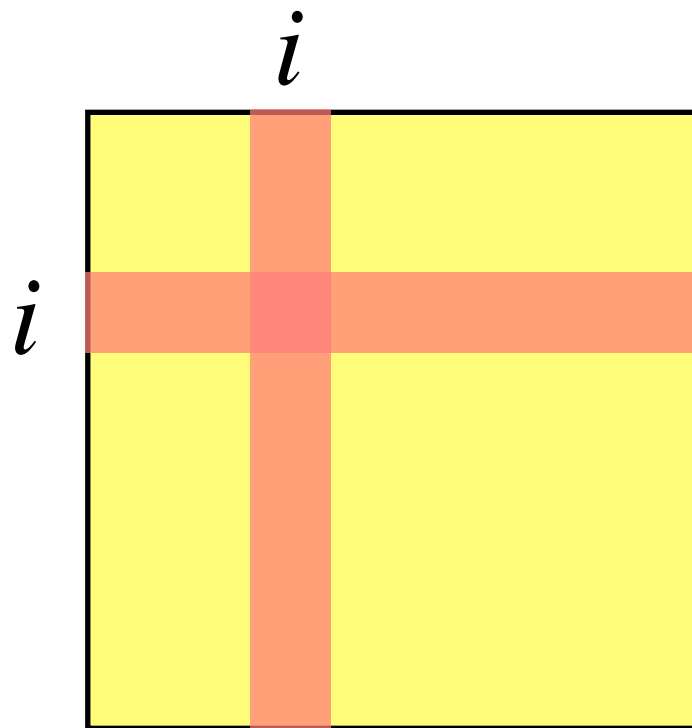
$$i \in V, e \in E \quad B(i, e) = \begin{cases} 1 & e = \{i, j\}, i < j \\ -1 & e = \{i, j\}, i > j \\ 0 & \text{otherwise} \end{cases}$$



$$L = BB^T$$

Kirchhoff's matrix-tree theorem

$L_{i,i}$: submatrix of L obtained by removing the i th row and i th column



$t(G)$: number of spanning trees in G

Kirchhoff's matrix-tree theorem

$L_{i,i}$: submatrix of L obtained by removing
the i th row and i th column

$t(G)$: number of spanning trees in G

Kirchhoff's Matrix-Tree Theorem:

$$\forall i, \quad t(G) = \det(L_{i,i})$$

Kirchhoff's Matrix-Tree Theorem:

$$\forall i, \quad t(G) = \det(L_{i,i})$$

$$B_i : (n - 1) \times m$$

incidence matrix B removing i th row

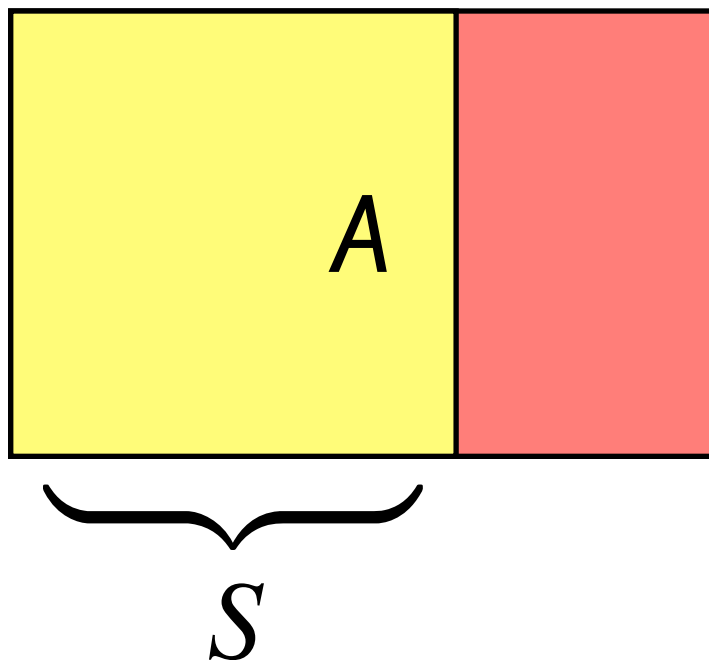
$$L = BB^T$$

$$L_{i,i} = B_i B_i^T \quad \det(L_{i,i}) = \det(B_i B_i^T) = ?$$

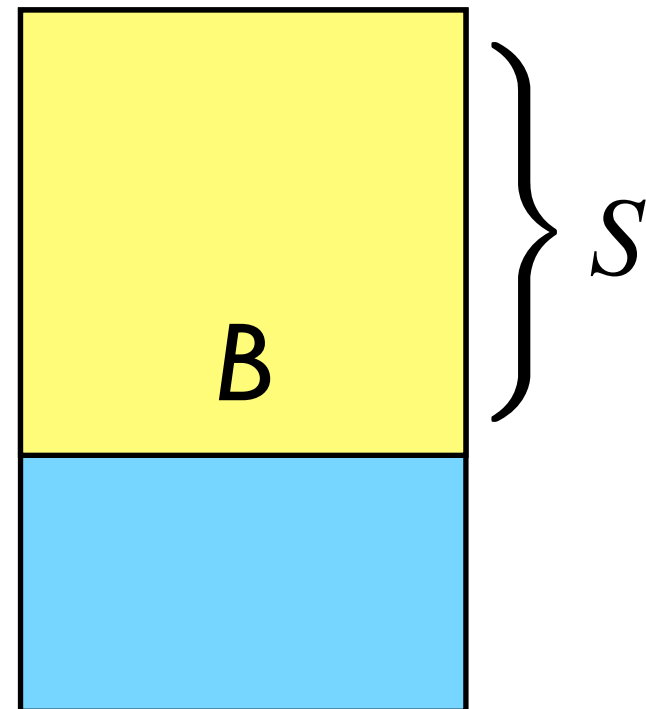
Cauchy-Binet Theorem:

$$\det(AB) = \sum_{S \in \binom{[m]}{n}} \det(A_{[n], S}) \det(B_{S, [n]})$$

$A : n \times m$



$B : m \times n$



Cauchy-Binet Theorem:

$$\det(AB) = \sum_{S \in \binom{[m]}{n}} \det(A_{[n],S}) \det(B_{S,[n]})$$

$$\det(L_{i,i}) = \det(B_i B_i^T)$$

$$= \sum_{S \in \binom{[m]}{n-1}} \det(B_{[n] \setminus \{i\}, S}) \det(B_{S, [n] \setminus \{i\}}^T)$$

$$= \sum_{S \in \binom{[m]}{n-1}} \det(B_{[n] \setminus \{i\}, S})^2$$

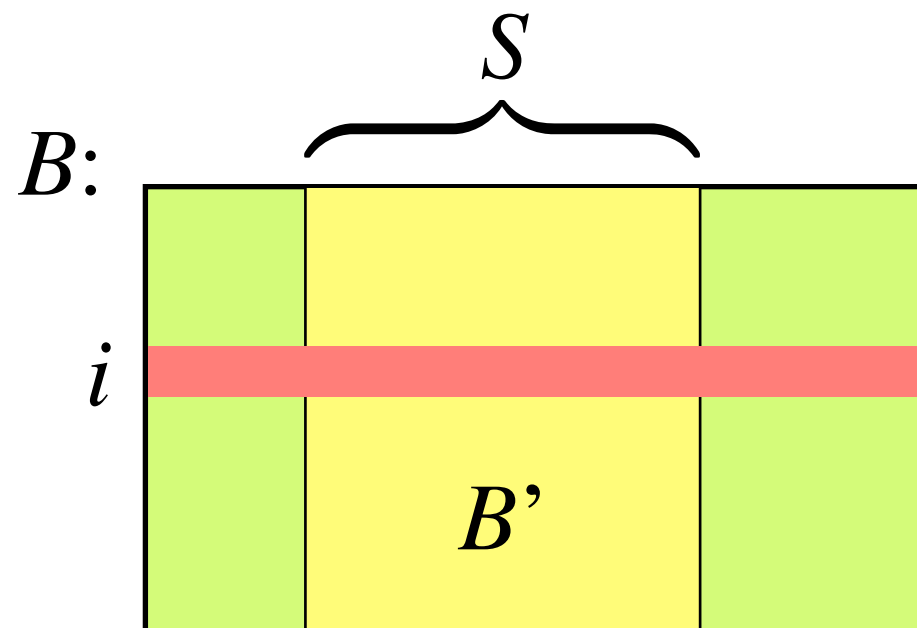
$$\det(L_{i,i}) = \sum_{S \in \binom{[m]}{n-1}} \det(B_{[n] \setminus \{i\}, S})^2$$

$$j \in [n] \setminus \{i\}, e \in S$$

$$B_{[n] \setminus \{i\}, S}(j, e) = \begin{cases} 1 & e = \{j, k\}, j < k \\ -1 & e = \{j, k\}, j > k \\ 0 & \text{otherwise} \end{cases}$$

$$\det(B_{[n] \setminus \{i\}, S}) = \begin{cases} \pm 1 & S \text{ is a spanning tree of } G \\ 0 & \text{otherwise} \end{cases}$$

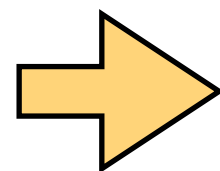
$$\det(B_{[n]\setminus\{i\},S}) = \begin{cases} \pm 1 & S \text{ is a spanning tree of } G \\ 0 & \text{otherwise} \end{cases}$$



$$B' = B_{[n]\setminus\{i\},S}$$

$(n-1) \times (n-1)$ matrix:

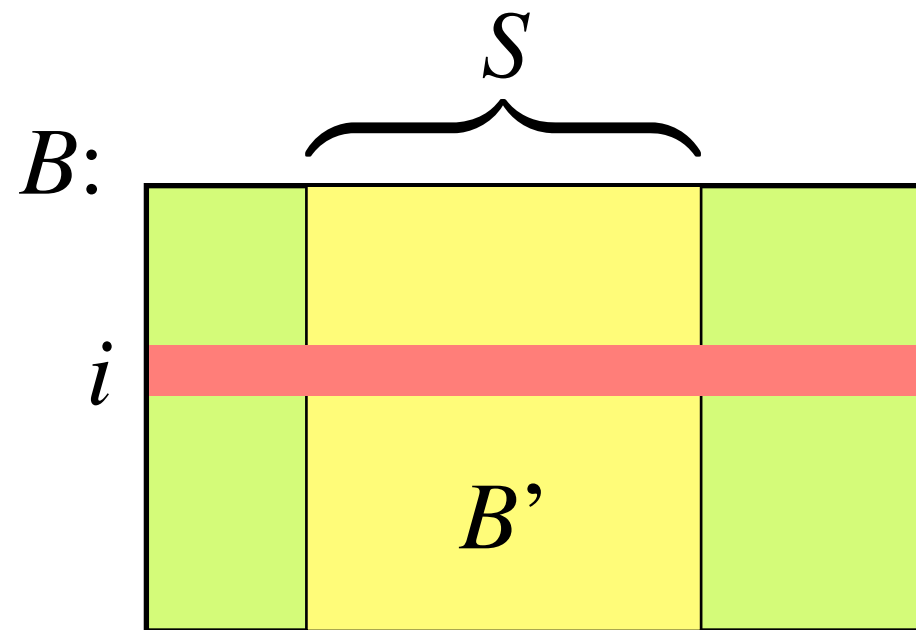
every column contains
at most one 1 and at most one -1
and all other entries are 0



$$\det(B') \in \{-1, 0, 1\}$$

$\det(B') \neq 0$ iff S is a spanning tree

$\det(B') \neq 0$ iff S is a spanning tree



S is not a spanning tree:

\exists a connected component R
s.t. $i \notin R$

$\Rightarrow \det(B') = 0$

S is a spanning tree:

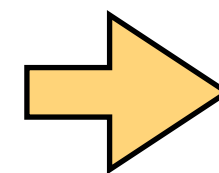
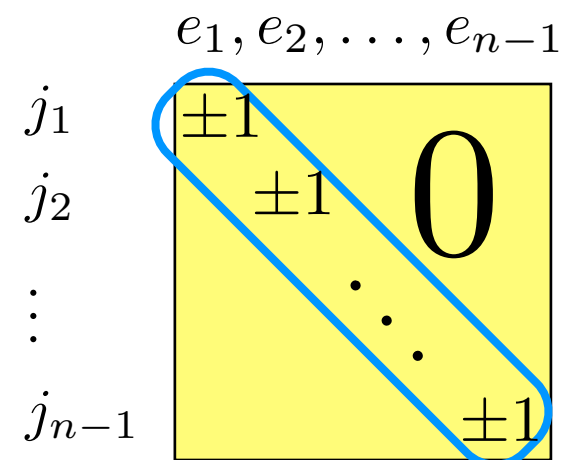
\exists a leaf $j_1 \neq i$ with incident edge e_1 , delete e_1

\exists a leaf $j_2 \neq i$ with incident edge e_2 , delete e_2

\vdots

vertices: j_1, j_2, \dots, j_{n-1}

edges: e_1, e_2, \dots, e_{n-1}



$\det(B') = \pm 1$

Cauchy-Binet

$$\det(L_{i,i}) = \sum_{S \in \binom{[m]}{n-1}} \det(B_{[n] \setminus \{i\}, S})^2$$

$$j \in [n] \setminus \{i\}, e \in S$$

$$B_{[n] \setminus \{i\}, S}(j, e) = \begin{cases} 1 & e = \{j, k\}, j < k \\ -1 & e = \{j, k\}, j > k \\ 0 & \text{otherwise} \end{cases}$$

$$\det(B_{[n] \setminus \{i\}, S}) = \begin{cases} \pm 1 & S \text{ is a spanning tree of } G \\ 0 & \text{otherwise} \end{cases}$$

Kirchhoff's Matrix-Tree Theorem:

$$\forall i, \quad t(G) = \det(L_{i,i})$$

all n -vertex trees: spanning trees of K_n

$$L_{i,i} = \begin{bmatrix} n-1 & -1 & \cdots & -1 \\ -1 & n-1 & \cdots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & \cdots & n-1 \end{bmatrix}$$

Cayley formula:

$$T_n = t(K_n) = \det(L_{i,i}) = n^{n-2}$$