Agenda

1. Counting formulas for complete graphs
2. Working with factorials
3. Counting spanning trees in networks
4. Kruskal's algorithm

Note: Graph coloring (the "Mini-Excursion") will **not** be on Monday's test.

1. Formulas for Complete Graphs

In the complete graph K_N:

- Number of edges in the whole graph: $N(N-1)/2$
- Number of edges in a spanning tree: $N-1$
- Number of Hamilton circuits: $(N-1)!$
- Number of spanning trees: N^{N-2} (Cayley's formula)

1. Formulas for Complete Graphs

Number of edges in $K_N = N(N-1)/2$.

- This is the same as the formula for the number of pairwise comparisons among N candidates.

1. Formulas for Complete Graphs

Number of edges in a spanning tree of $K_N = N - 1$

- This is because in **any** connected graph with N vertices, a spanning tree must have exactly $N - 1$ edges.

1. Formulas for Complete Graphs

Number of Hamilton circuits in $K_N = (N-1)!$

- After choosing the reference vertex, there are $N-1$ possibilities for the second vertex, $N-2$ possibilities for the third vertex, $\ldots$, 2 possibilities for the second-to-last vertex, 1 possibility for the last vertex.
- Multiplying all these numbers together gives $(N-1)(N-2)\cdots(2)(1) = (N-1)!$

1. Formulas for Complete Graphs

Number of spanning trees in $K_N = N^{N-2}$.

- We worked several examples out in class:

  K_1: 1 spanning tree ($= 1^{-1}$)

  K_2: 1 spanning tree ($= 2^0$)

  K_3: 3 spanning trees ($= 3^1$)

  K_4: 16 spanning trees ($= 4^2$)

  K_5: 125 spanning trees ($= 5^3$)

  …

  K_N: N^{N-2} spanning trees

2. Factorials

$$
\begin{aligned}
1! &= 1 = \mathbf{1} \\
2! &= 1 \times 2 = \mathbf{2} \\
3! &= 1 \times 2 \times 3 = \mathbf{6} \\
4! &= 1 \times 2 \times 3 \times 4 = \mathbf{24} \\
5! &= 1 \times 2 \times 3 \times 4 \times 5 = \mathbf{120} \\
&\vdots \\
12! &= 1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8 \times 9 \times 10 \times 11 \times 12 \\
&= \mathbf{479001600} \\
13! &= 1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8 \times 9 \times 10 \times 11 \times 12 \times 13 \\
&= \mathbf{6227020800}
\end{aligned}
$$

2. Factorials

$70! =$
$1197857166996989179607278372168909873645893814254642585755536286462800958278984531968000000000000000$

This is too many digits for most calculators.

Even if your calculator could count this high, it would probably display it as something like

$$1.1978571669\text{E}100.$$

2. Factorials

Problem #1: Simplify $4!/3!$ without using a calculator.

Problem #2: Simplify $13!/12!$ without using a calculator.

2. Factorials

Problem #1: Simplify $4!/3!$ without using a calculator.

$$\begin{aligned} 3! &= 1 \times 2 \times 3 = 6 \\ 4! &= 1 \times 2 \times 3 \times 4 = 24 \\ 4!/3! &= 24/6 = 4 \end{aligned}$$

But also,

$$\frac{4!}{3!} = \frac{1 \times 2 \times 3 \times 4}{1 \times 2 \times 3} = 4.$$

2. Factorials

Similarly,

$$\frac{13!}{12!} = 6227020800/479001600 \quad \text{(yuck)}$$

$$= \frac{1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8 \times 9 \times 10 \times 11 \times 12 \times 13}{1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8 \times 9 \times 10 \times 11 \times 12}$$

$$= 13.$$

In general,

$$\boxed{\frac{\mathbf{N!}}{\mathbf{(N-1)!}} = \mathbf{N.}}$$

2. Factorials

Similarly,

$$\frac{13!}{12!} = 6227020800/479001600 \quad \text{(yuck)}$$

$$= \frac{1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8 \times 9 \times 10 \times 11 \times 12 \times 13}{1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8 \times 9 \times 10 \times 11 \times 12}$$

$$= 13.$$

In general,

$$\boxed{\frac{\mathbf{N!}}{\mathbf{(N-1)!}} = \mathbf{N.}}$$

2. Factorials

Sample problems:

$$\frac{6!}{5!} = \square$$

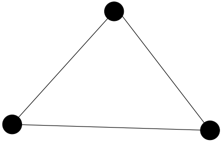
$$\frac{100!}{98!} = \square$$

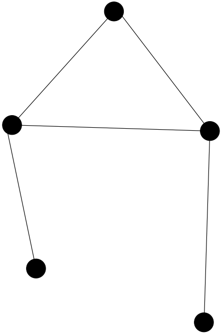
$$\frac{16!}{15!} = \square$$

$$\frac{71!}{70!} = \square$$

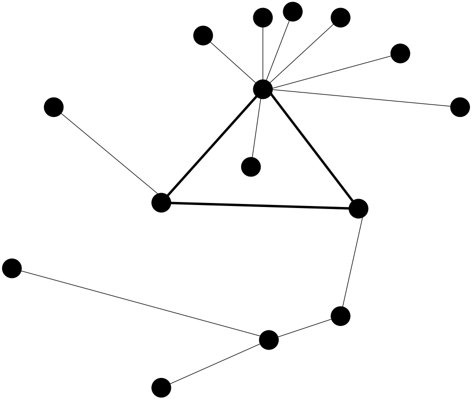
$$\frac{18!}{17!} - \frac{16!}{15!} - \frac{14!}{13!} + \frac{12!}{11!} = \square$$

3. Counting Spanning Trees in Networks

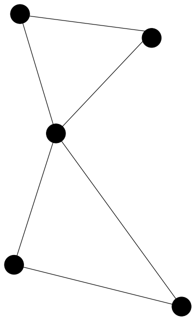


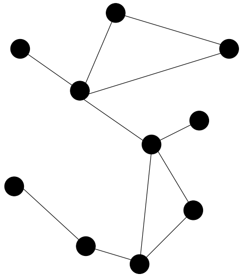


3. Counting Spanning Trees in Networks

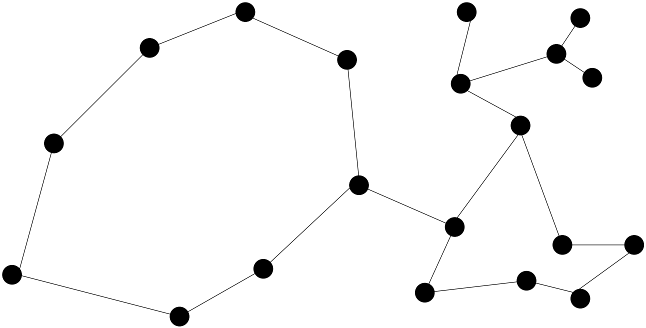


3. Counting Spanning Trees in Networks

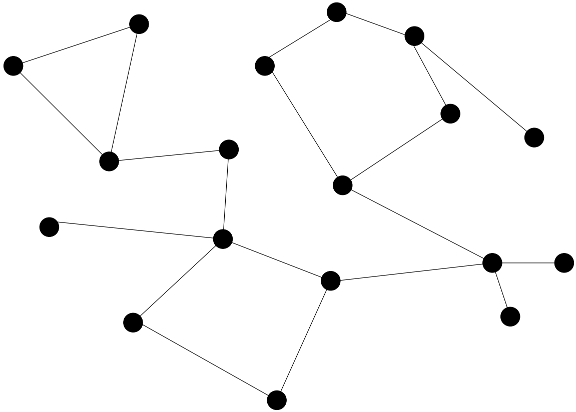




3. Counting Spanning Trees in Networks



3. Counting Spanning Trees in Networks



3. Counting Spanning Trees in Networks

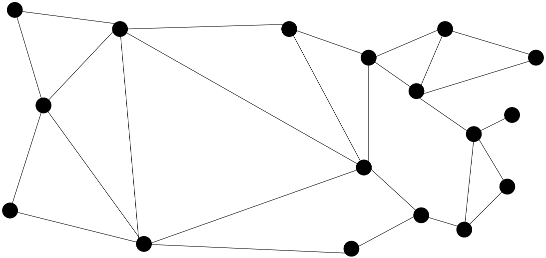
Summary: If a network consists of a bunch of **circuits** attached to each other by **bridges**,

then the **number of spanning trees** is the **product of the lengths of the circuits**.

The other kind of network for which we can easily count trees is a **complete graph** $\mathbf{K_N}$, where there are $\mathbf{N^{N-2}}$ spanning trees. T

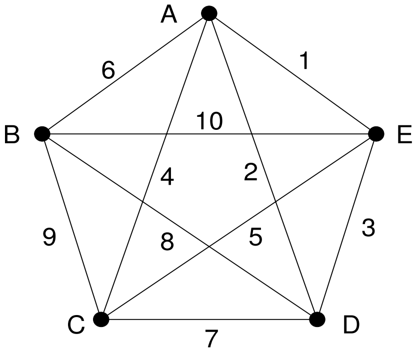
3. Counting Spanning Trees in Networks

Many networks are not of this form — the circuits might overlap.



Here, it is very hard to count the spanning trees.

4. Kruskal's Algorithm



Weights of edges

	A	B	C	D	E
A		6	4	2	1
B	6		9	8	10
C	4	9		7	5
D	2	8	7		3
E	1	10	5	3	