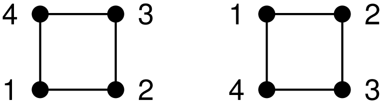
Chapter 3: Graph Isomorphism

Consider the two graphs below. Are they *equal*?



Yes: Identical vertex sets and edge sets

What about these two?

No: 34 is an edge in the first but not the second

They sure look like close cousins... We already discussed a word to represent the relationship between the two graphs above.

Isomorphic graphs

Definition

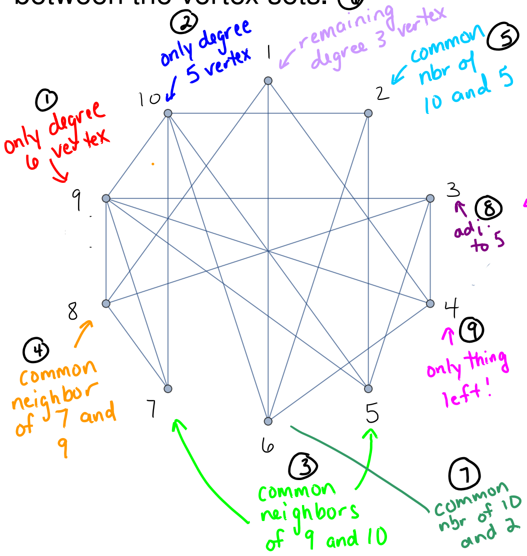
Two labeled graphs G and H are *isomorphic* if there exists a bijection $\phi : V(G) \to V(H)$ such that $uv \in E(G)$ if and only if $\phi(u)\phi(v) \in E(H)$.

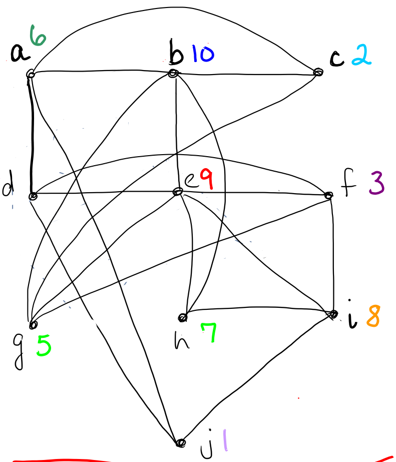
perfect match

- Remember, a bijection is one-to-one and onto; also known as a *one-to-one correspondence*.
- Rephrased: We need to be able to relabel the graph H with the vertex labels on G so that we get a graph *equal* to G.
- Two *unlabeled* graphs are isomorphic if they can be labeled so that they are equal.
- A more geometric notion: Two graphs are isomorphic if we can move the vertices and edges in space (without changing any adjacencies) so that we get the same picture.

Example

Show that the following graphs are isomorphic by providing a bijection between the vertex sets. 6

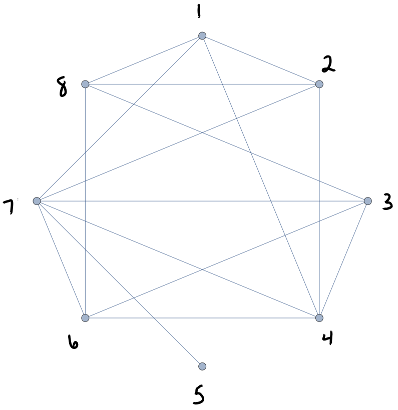


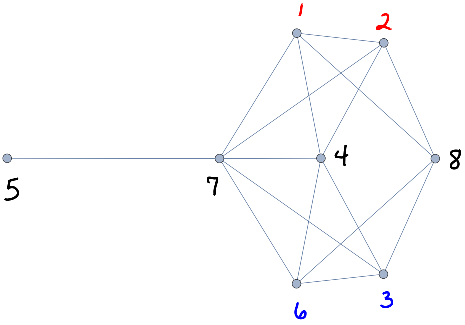


Bijection: $\phi = \{(1,j),(2,c),(3,f),(4,d),(5,g),(6,a),(7,h),(8,i),(9,e),(10,b)\}$

Example

Show that the following graphs are isomorphic by providing an appropriate labeling of the vertices.





* labelling is not unique due to symmetry

1 can be swapped with 2
6 can be swapped with 3
or the pairs can swap with each other

Isomorphism classes

Definition

The isomor~~phic~~phism class for a graph G is the set of all graphs isomorphic to G.

- We have already been introduced to this idea - when I say K_5 is "the" complete graph with 5 vertices, what I really mean is that K_5 represents the *isomorphism class* of all complete graphs with 5 vertices.
- From a mathematicians point of view, isomorphic graphs are essentially equal.
- In applications, it is important to be able to make a distinction between *equal* graphs and *isomorphic* graphs.

Example

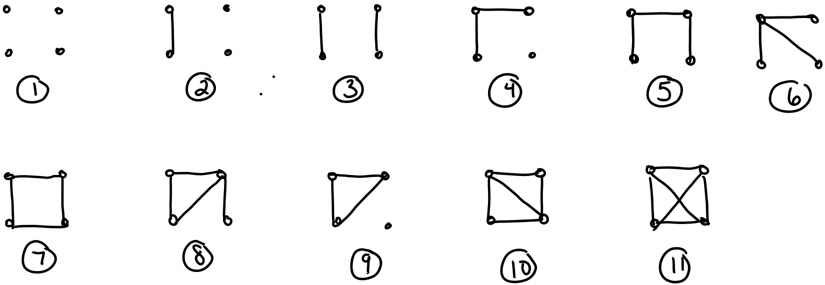
Draw all non-isomorphic graphs of order 3.

That's all!

Up to $n=19$ can be found here, no formula is known:

https://oeis.org/A000088

Order 4:



Non-isomorphic graphs

Question: How do we show graphs are NOT isomorphic?

Definition

An *isomorphic invariant* of a graph is a property P for which the following is true for every pair of graphs G and H:

"If G and H are isomorphic and G has property P, then H must also have property P."

- An easy example: $P =$ the order of the graph
- We can use this to argue that two graphs are *not* isomorphic: "Since G has 5 vertices and H has 6 vertices, G and H are not isomorphic."

Isomorphic Invariants

A list of isomorphic invariants:

"neighborhood structure"

order

size

degree sequence / degree sum / being regular

being bipartite / tripartite / k-partite

having a cycle of length —

having a path of length —

having a complete subgraph with — vertices

containing any subgraph

diameter

# of components

$\Delta(G)$, $\delta(G)$

diameter

containing Hamilton cycle / Euler circuit

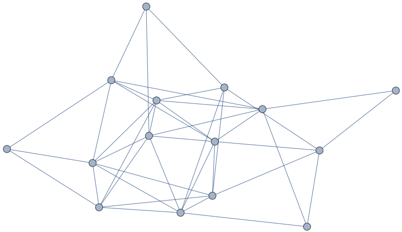
being planar

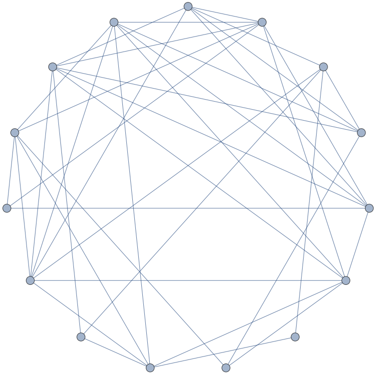
A BIG problem

- Although we can use an isomorphic invariant to show that two graphs are *not* isomorphic (if we find the right invariant!), we CANNOT use them to argue that G and H *are* isomorphic.
- More precisely: Let L be an finite list of isomorphic invariants. Then there exist graphs G and H such that G and H share each property in L, but G and H are NOT isomorphic.
- To show that G and H *are* isomorphic, we *must produce* an isomorphism between the vertices.

IMPORTANT!

Are these two graphs isomorphic?





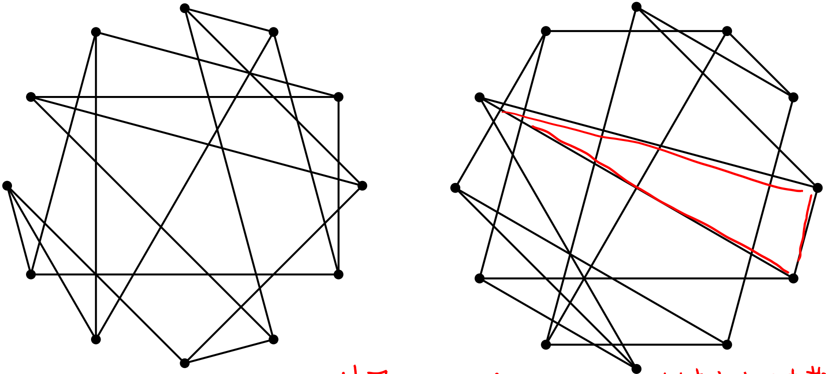
I happen to know they are, because I generated them that way, but I wouldn't want to check!

More bad news

- If we want to argue that G and H are *not* isomorphic and we can't find a nice isomorphic invariant to use, we may need to check **all possible maps** between the vertex sets and show that none of them preserves the edge structure.
- In two 50-vertex graphs, there are 3.04×10^{64} different maps between them!!!!

(In practice, there are ways to speed this up - this is a worst-case scenario.)

Are these two graphs isomorphic?



Wait, we have been saved! There is a triangle in the right but not the left.

I don't know! I had Mathematica generate two 3-regular graphs at random. There are 85 of them on 12 vertices, so probably not, but I don't know for sure. This problem is *hard*!

A note about complexity

The P vs. NP problem:

- Suppose I have a problem for which it is “easy” to verify that a solution is correct.
  - For example: If I claim to have found a bijection between the vertex sets of two graphs, it is “easy” to verify whether it is correct.
  - “Easy” means “”there is an algorithm that will do it whose running time is polynomial in the size of the input” (like $O(n^2)$).
- Must there also exist a polynomial algorithm to solve the problem?
- The current answer: Who knows?
- What is known: There is a VERY large class of problems, called the NP-Complete problems, that have been shown to have the same answer - that is, if a polynomial-time algorithm exists to solve ONE of them, then a polynomial-time algorithm exists to solve ALL of them.

Back to the graph isomorphism problem

- The graph isomorphism problem is unusual among many computational problems: It is unknown whether or not it is NP-complete.
- Until just a few years ago, the best known algorithm, due to Babai and Luks (1983) ran in $\mathbf{2}^{O(\sqrt{n}\log n)}$ time, where n is the number of vertices.
- Recently (and amazingly!) an algorithm that runs in $\mathbf{2}^{O((\log n)^3)}$ was announced by Babai.
- The proof is still under review...
- In practice, there are fast algorithms that "usually" do better, but in their worst case they can be very, very bad.