Discrete Mathematics for Computing

UTD

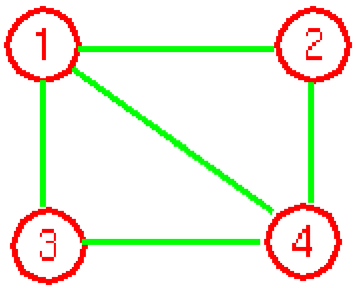
Ch 10.3 Representing Graphs and Graph Isomorphism

- Discussion
- Have you seen graphs before?
- If yes, what types?
- Can graphs have different structures, yet be identical in some way?

Chemical Compounds

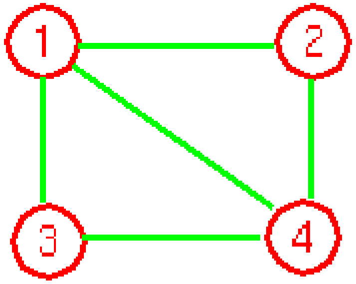
Adjacency Lists

- Used to represent graphs with no multiple edges
- Specify which vertices are adjacent to each vertex



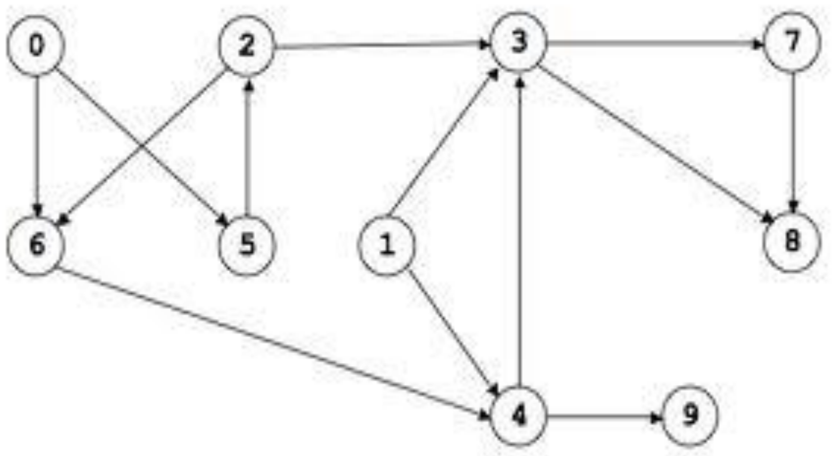
Adjacency Lists

Vertex	Adjacent Vertices
1	2, 3, 4
2	1, 4
3	1, 4
4	1, 2, 3



Adjacency Lists

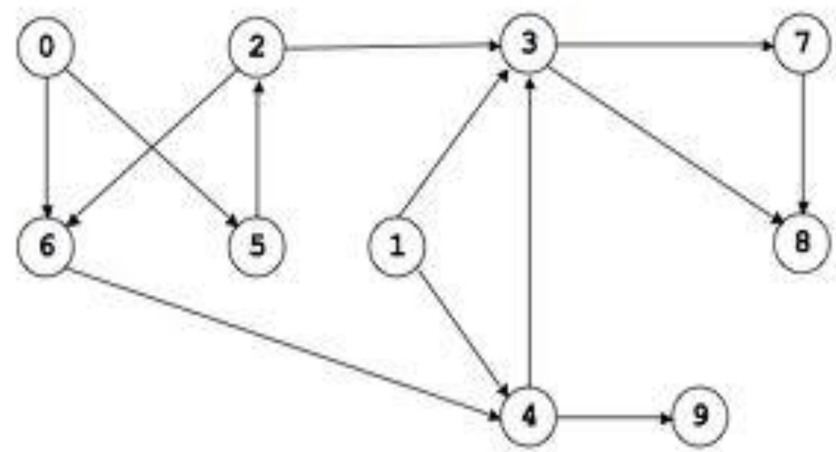
- Consider the below digraph
- Initial and terminal vertices



Adjacency Lists

-

InitialVertex	Terminal Vertices
0	5, 6
1	3, 4
2	3, 6
3	7, 8
4	3, 9
5	2
6	4
7	8
8	
9	

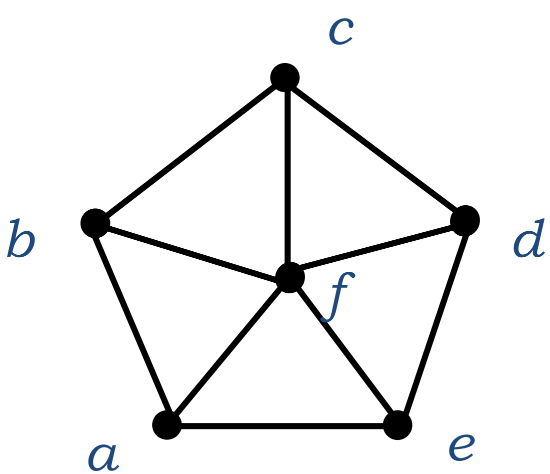


Adjacency Matrix

A simple graph $G = (V,E)$ with n vertices can be represented by its *adjacency matrix*, A, where the entry a_{ij} in row i and column j is:

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \text{ is an edge in } G \\ 0 & \text{otherwise} \end{cases}$$

Adjacency Matrix Example



W_5

$\{v_1, v_2\}$

row column

$$
\begin{array}{c c}
 & \text{To} \\
 & \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\
\text{From} & \\
\begin{array}{c} a \\ b \\ c \\ d \\ e \\ f \end{array} &
\begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 \\
1 & 0 & 0 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 0
\end{bmatrix}
\end{array}
$$

Tradeoffs

- Simple Graphs – relatively few edges

  SPARSE

  Adjacency Lists
- Graphs – many edges, more than half of all possible edges

  DENSE

  Adjacency Matrices

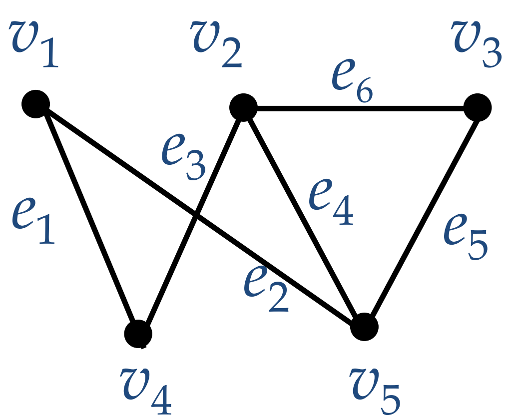
Incidence Matrix

- Let $G = (V,E)$ be an undirected graph. Suppose $v_1,v_2,v_3,\ldots,v_n$ are the vertices and $e_1,e_2,e_3,\ldots,e_m$ are the edges of G. The *incidence matrix* w.r.t. this ordering of V and E is the $n\times m$ matrix $M = [m_{ij}]$, where

$$m_{ij} = \begin{cases} 1 & \text{if edge } e_j \text{ is incident with } v_i \\ 0 & \text{otherwise} \end{cases}$$

Incidence Matrix Example

- Represent the graph shown with an incidence matrix.



	e_1	e_2	e_3	e_4	e_5	e_6
v_1	1	1	0	0	0	0
v_2	0	0	1	1	0	1
v_3	0	0	0	0	1	1
v_4	1	0	1	0	0	0
v_5	0	1	0	1	1	0

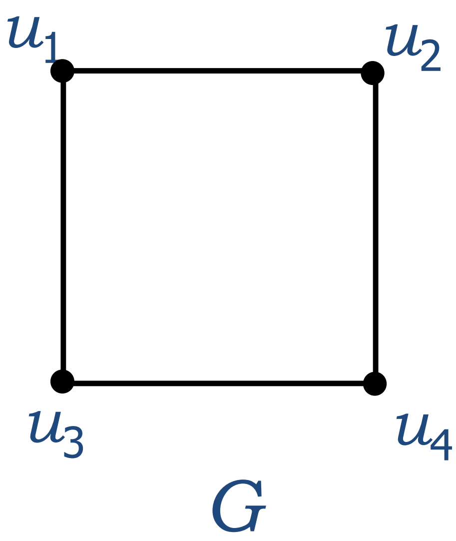
← edges

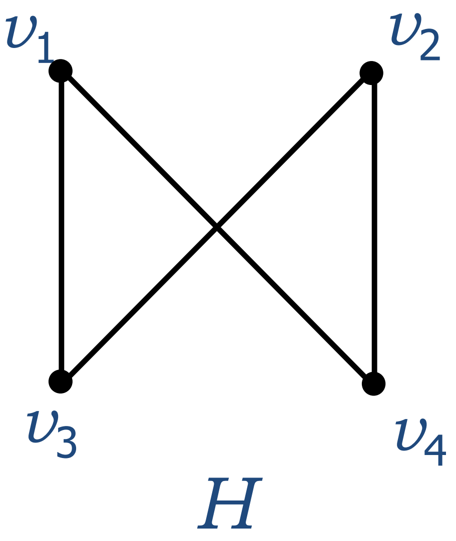
↑ vertices

Isomorphism

- Two simple graphs are isomorphic if:
  - there is a one-to one correspondence between the vertices of the two graphs
  - the adjacency relationship is preserved

Example





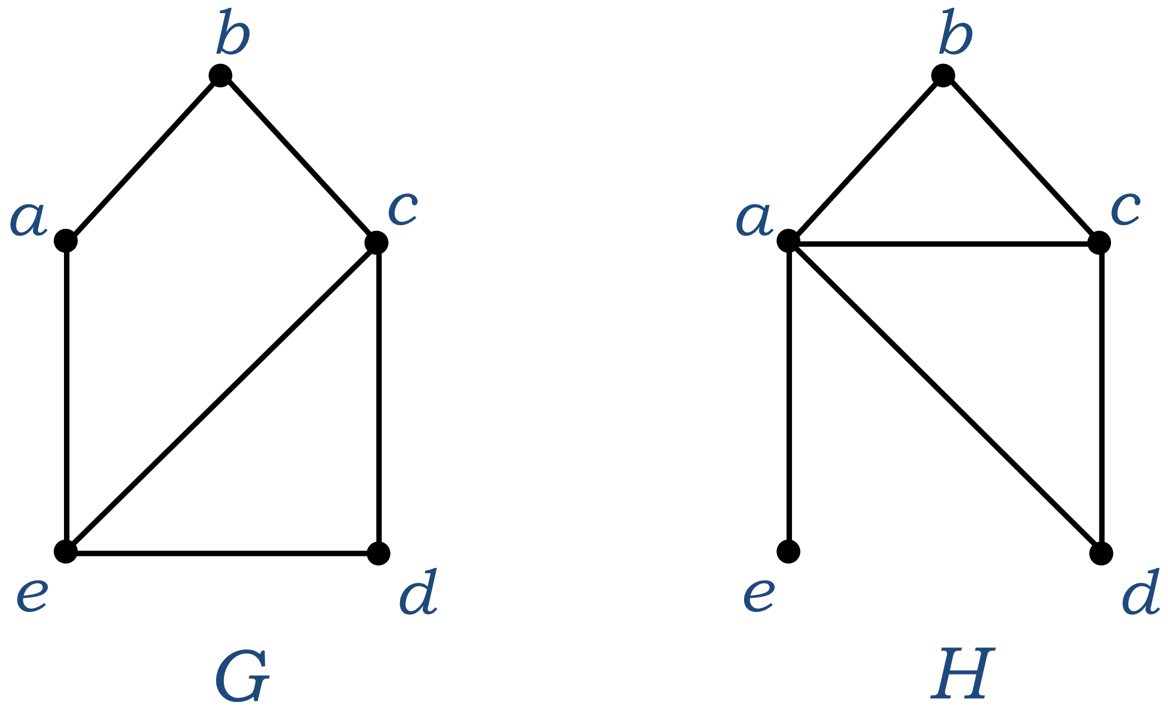
Are G and H isomorphic?

$$f(u_1) = v_1,\; f(u_2) = v_4,\; f(u_3) = v_3,\; f(u_4) = v_2$$

Invariants

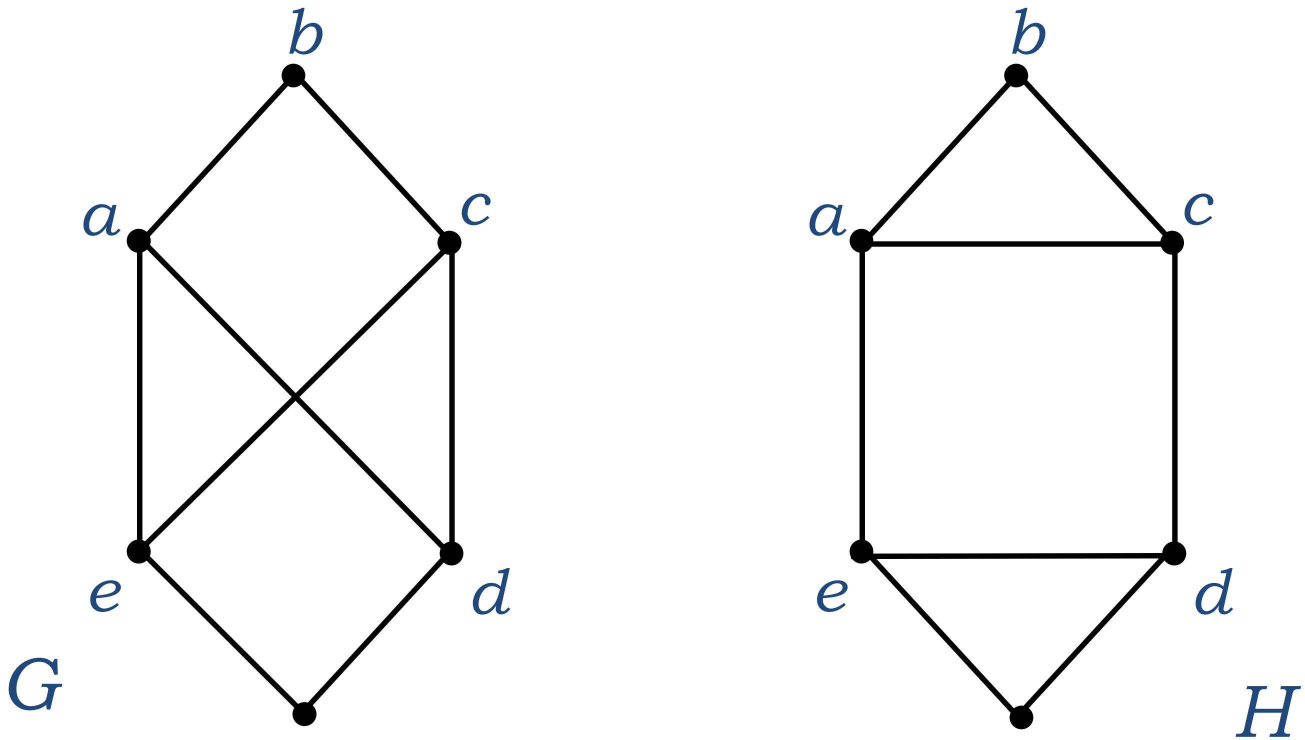
- Invariants – properties that two simple graphs must have in common to be isomorphic
  - Same number of vertices
  - Same number of edges
  - Degrees of corresponding vertices are the same
  - If one is bipartite, the other must be; if one is complete, the other must be; and others …

Example



Are G and H isomorphic?

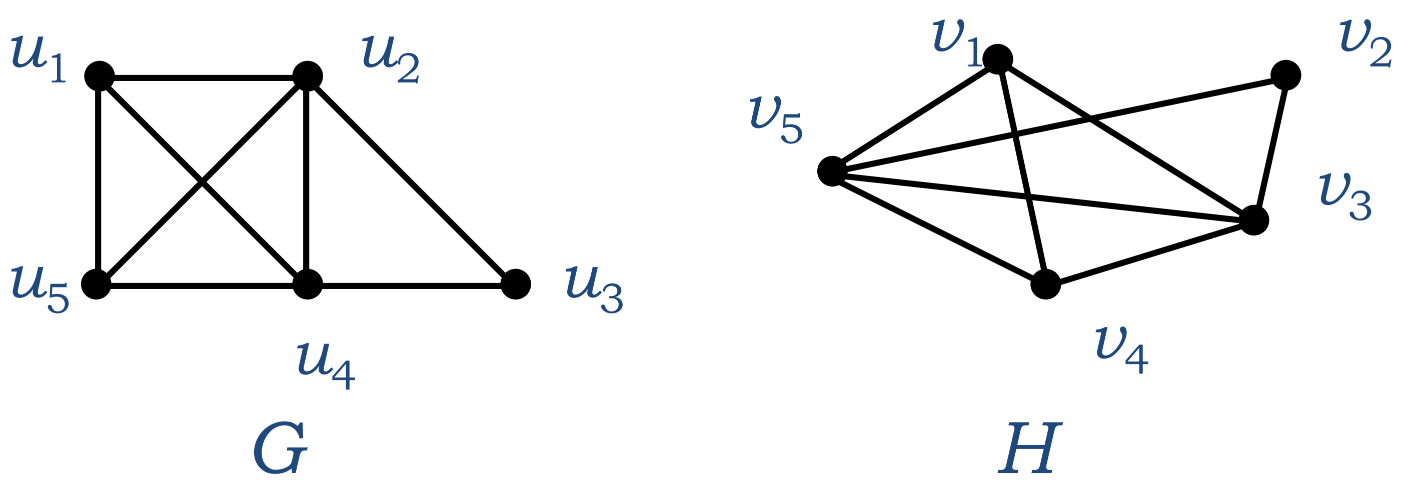
Example



Are G and H isomorphic?

Example

- Are these two graphs isomorphic?



  - They both have 5 vertices
  - They both have 8 edges
  - They have the same number of vertices with the same degrees: 2, 3, 3, 4, 4.

Example (Cont.)

$$
\begin{array}{c}
\text{G} \\
\begin{array}{c|ccccc}
 & u_1 & u_2 & u_3 & u_4 & u_5 \\
\hline
u_1 & 0 & 1 & 0 & 1 & 1 \\
u_2 & 1 & 0 & 1 & 1 & 1 \\
u_3 & 0 & 1 & 0 & 1 & 0 \\
u_4 & 1 & 1 & 1 & 0 & 1 \\
u_5 & 1 & 1 & 0 & 1 & 0
\end{array}
\end{array}
\qquad
\begin{array}{c}
\text{H} \\
\begin{array}{c|ccccc}
 & v_1 & v_2 & v_3 & v_4 & v_5 \\
\hline
v_1 & 0 & 0 & 1 & 1 & 1 \\
v_2 & 0 & 0 & 1 & 0 & 1 \\
v_3 & 1 & 1 & 0 & 1 & 1 \\
v_4 & 1 & 0 & 1 & 0 & 1 \\
v_5 & 1 & 1 & 1 & 1 & 0
\end{array}
\end{array}
\qquad
\begin{array}{c}
\text{H} \rightarrow \text{G?} \\
\begin{array}{c|ccccc}
 & v_1 & v_3 & v_2 & v_5 & v_4 \\
\hline
v_1 & & & & & \\
v_3 & & & & & \\
v_2 & & & & & \\
v_5 & & & & & \\
v_4 & & & & &
\end{array}
\end{array}
$$

– G and H don't appear to be isomorphic.

– However, we haven't tried mapping vertices from G onto H yet.

Example (Cont.)

- Start with the vertices of degree 2 since each graph only has one:

$$\deg(u_3) = \deg(v_2) = 2 \quad \text{therefore} \quad f(u_3) = v_2$$

UTD

Example (Cont.)

- Now consider vertices of degree 3

$$\deg(u_1) = \deg(u_5) = \deg(v_1) = \deg(v_4) = 3$$

therefore we must have either one of

$$f(u_1) = v_1 \text{ and } f(u_5) = v_4$$

$$f(u_1) = v_4 \text{ and } f(u_5) = v_1$$

Example (Cont.)

- Now try vertices of degree 4:

  $\deg(u_2) = \deg(u_4) = \deg(v_3) = \deg(v_5) = 4$

  therefore we must have one of:

  $f(u_2) = v_3$ and $f(u_4) = v_5$ or

  $f(u_2) = v_5$ and $f(u_4) = v_3$

Example (Cont.)

- There are four possibilities (this can get messy!)

$$f(u_1) = v_1,\ f(u_2) = v_3,\ f(u_3) = v_2,\ f(u_4) = v_5,\ f(u_5) = v_4$$

$$f(u_1) = v_4,\ f(u_2) = v_3,\ f(u_3) = v_2,\ f(u_4) = v_5,\ f(u_5) = v_1$$

$$f(u_1) = v_1,\ f(u_2) = v_5,\ f(u_3) = v_2,\ f(u_4) = v_3,\ f(u_5) = v_4$$

$$f(u_1) = v_4,\ f(u_2) = v_5,\ f(u_3) = v_2,\ f(u_4) = v_3,\ f(u_5) = v_1$$

Example (Cont.)

G H H'

$$\begin{array}{c|ccccc} & u_1 & u_2 & u_3 & u_4 & u_5 \\ \hline u_1 & 0 & 1 & 0 & 1 & 1 \\ u_2 & 1 & 0 & 1 & 1 & 1 \\ u_3 & 0 & 1 & 0 & 1 & 0 \\ u_4 & 1 & 1 & 1 & 0 & 1 \\ u_5 & 1 & 1 & 0 & 1 & 0 \end{array} \quad \begin{array}{c|ccccc} & v_1 & v_2 & v_3 & v_4 & v_5 \\ \hline v_1 & 0 & 0 & 1 & 1 & 1 \\ v_2 & 0 & 0 & 1 & 0 & 1 \\ v_3 & 1 & 1 & 0 & 1 & 1 \\ v_4 & 1 & 0 & 1 & 0 & 1 \\ v_5 & 1 & 1 & 1 & 1 & 0 \end{array} \quad \begin{array}{c|ccccc} & v_1 & v_3 & v_2 & v_5 & v_4 \\ \hline v_1 & 0 & 1 & 0 & 1 & 1 \\ v_3 & 1 & 0 & 1 & 1 & 1 \\ v_2 & 0 & 1 & 0 & 1 & 0 \\ v_5 & 1 & 1 & 1 & 0 & 1 \\ v_4 & 1 & 1 & 0 & 1 & 0 \end{array}$$

We permute the adjacency matrix of H (per function choices above) to see if we get the adjacency of G. Let's try:

$$f(u_1) = v_1, f(u_2) = v_3, f(u_3) = v_2, f(u_4) = v_5, f(u_5) = v_4$$

Does G = H'? Yes!

Example (Cont.)

G H H'

$$\begin{array}{c|ccccc} & u_1 & u_2 & u_3 & u_4 & u_5 \\ \hline u_1 & 0 & 1 & 0 & 1 & 1 \\ u_2 & 1 & 0 & 1 & 1 & 1 \\ u_3 & 0 & 1 & 0 & 1 & 0 \\ u_4 & 1 & 1 & 1 & 0 & 1 \\ u_5 & 1 & 1 & 0 & 1 & 0 \end{array} \quad \begin{array}{c|ccccc} & v_1 & v_2 & v_3 & v_4 & v_5 \\ \hline v_1 & 0 & 0 & 1 & 1 & 1 \\ v_2 & 0 & 0 & 1 & 0 & 1 \\ v_3 & 1 & 1 & 0 & 1 & 1 \\ v_4 & 1 & 0 & 1 & 0 & 1 \\ v_5 & 1 & 1 & 1 & 1 & 0 \end{array} \quad \begin{array}{c|ccccc} & v_4 & v_3 & v_2 & v_5 & v_1 \\ \hline v_4 & 0 & 1 & 0 & 1 & 1 \\ v_3 & 1 & 0 & 1 & 1 & 1 \\ v_2 & 0 & 1 & 0 & 1 & 0 \\ v_5 & 1 & 1 & 1 & 0 & 1 \\ v_1 & 1 & 1 & 0 & 1 & 0 \end{array}$$

It turns out that

$$f(u_1) = v_4, f(u_2) = v_3, f(u_3) = v_2, f(u_4) = v_5, f(u_5) = v_1$$

also works.

Algorithms for Graph Isomorphism

- The best algorithms known for determining whether two graphs are isomorphic have exponential worst-case time complexity (in the number of vertices of the graphs).
- However, there are algorithms with linear average-case time complexity.
- You can use a public domain program called NAUTY to determine in less than a second whether two graphs with as many as 100 vertices are isomoprhic.

UTD

Applications of Graph Isomorphism

- Chemists use molecular graphs to model chemical compounds. Vertices represent atoms and edges represent chemical bonds. When a new compound is synthesized, a database of molecular graphs is checked to determine whether the graph representing the new compound is isomorphic to the graph of a compound that this already known.
- Electronic circuits are modeled as graphs in which the vertices represent components and the edges represent connections between them. Graph isomorphism is the basis for
  - the verification that a particular layout of a circuit corresponds to the design's original schematics.
  - determining whether a chip from one vendor includes the intellectual property of another vendor.