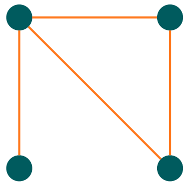
Graphs

A graph[1] is a set of nodes[2] V connected by a set of edges E.



The set V can be anything but the set E must be a set of 2-subsets of V. For example, V might be the set of cities in Ireland and E might represent motorways connecting cities. A motorway in this case would be defined by the two cities it connects.

[1] Michael Sipser. *Introduction to the Theory of Computation.* Third. Boston, MA: Course Technology, 2013. ISBN: 113318779X, p. 10.

[2] Nodes are sometimes also called vertices.

Set Notation

A set is a collection of objects, usually denoted using curly braces.[3] For example, the set A below contains the three objects 1, 2, and 3. The objects are usually called elements of the set.

$$A = \{1, 2, 3\}$$

Sets can be infinite, in which case the elements can be identified by an algorithm or property. In this case we usually assume the infinite set of counting numbers[4] $\mathbb{N}_0 = \{0, 1, 2, 3, \dots\}$ is a given[5].

In the below, the set T of even positive natural numbers is given by an algorithm. The algorithm says start with a natural number and multiply it by two. The set P is given by the property that each element is prime.

$$T = \{2n \mid n \in \mathbb{N}\}$$

$$P = \{p \in \mathbb{N} \mid p \text{ is prime}\}$$

Two important properties of sets are that sets are unordered and that each element is distinct. Note there is no mention of order in the definition *collection of objects*[6]. Likewise, the idea of an *object* is that it is unique — we did not say an instance of an object or anything like that.

We say B is a subset of A if all of the elements on B are also in A. When B has k elements, we sometimes say B is a k-subset of A. Under this definition, the empty set and A itself are always subsets of a set A.

Note that a set B is an object itself, and so might be an element of a set A. In this case, we are not saying that the elements of B are individually in A, although that could also be the case. The distinction is important[7].

[3] Sipser, *Introduction to the Theory of Computation*, p. 3.

[4] The numbers are usually called the natural numbers and $\mathbb{N}_0$ is the set of natural numbers including zero.

[5] Sometimes it is convenient to not include the element 0, in which case we denote the set $\mathbb{N}$.

[6] We can create an order or ordering of a set if we wish but that is something we must treat alongside the set.

[7] Bertrand Russel is known for Russell's paradox about a set R which is the set of all sets that do not contain themselves. A set seemingly may contain itself — consider the set of all sets. Does R contain itself?

Tuple Notation

When order matters, we use tuples. Tuples are basically the same as lists of arrays in programming languages.

A tuple is a finite sequence.[8] A sequence is a list of objects, usually stipulated to come from a set or sets. A tuple of length k is sometimes called a k-tuple, although a 2-tuple is usually just called a pair.

[8] Sipser, *Introduction to the Theory of Computation*, p. 6.

The word *list* implies an order — we can talk about the first thing on a list, if it exists.

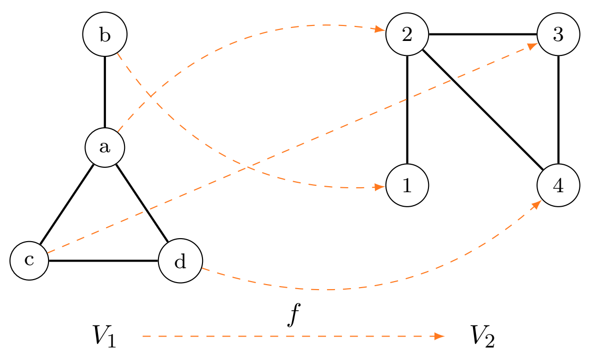
Graph Notation

A simple graph G is a pair $G = (V, E)$ where V is a set and E is a set of 2-subsets of V.

Isomorphism

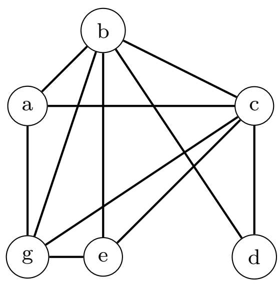
Graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Bijection $f : V_1 \to V_2$ such that $f(E_1) = E_2$ where $f(E_1) = \{\{f(v_1), f(v_2)\} | \{v_1, v_2\} \in E_1\}$.

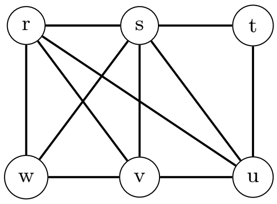
Example



$$
\begin{aligned}
f(E_1) \quad &= \{\{f(a), f(b)\}, \{f(a), f(c)\}, \\
&\quad \{f(a), f(d)\}, \{f(c), f(d)\}\} \\
&= \{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{3, 4\}\} = E_2
\end{aligned}
$$

Non-isomorphism





No of maps

$f(a) \to 6$ choices; $f(b) \to 5$ choices; $f(c) \to 4$ choices; etc. So, $n!$ maps between the vertex sets of two graphs with n vertices.

Some invariants

- Degrees.
- Paths.
- Connection.

Adjacency matrix

Fix a listing of V. $[a_{ij}]$ where a_{ij} is 1 if $\{v_i, v_j\} \in E$ else 0.

Example

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Permutation matrix

Isomorphic $\leftrightarrow \exists P$ such that $A = PBP^{\intercal}$.

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Binary encoding

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

$$\to 011001101111111011101100001100111101 0$$

$$\text{or } 111011011011101$$

Decision problem

$$f(110101, 101011) \to \text{Yes}$$

$$f(111011011011101, 1011111101110011) \to \text{No}$$

$$f : \{0,1\}^* \times \{0,1\}^* \to \{0,1\} = 1 \text{ iff isomorphic}$$

$$\mathbf{GRAPHISO} = \{(G_1, G_2) | f(G_1, G_2) = 1\}$$

References

Sipser, Michael. *Introduction to the Theory of Computation*. Third. Boston, MA: Course Technology, 2013. ISBN: 113318779X.