Why

We want to visualize relations.

Definition

A *directed graph* (or *digraph*, *graph*) is a pair (V, E) in which V is a nonempty set and E is a subset of $V \times V$. In other words, E is a relation on V. We call the elements of V *vertices* and the elements of E *edges* (or *arcs*).

Example

For example, define V and E by

$$V = \{1, 2, 3, 4\} \quad \text{and} \quad E = \{(1, 2), (1, 3), (2, 4), (3, 4)\}$$

It is worth drawing this graph.

Edge and vertex terminology

Let $(v, w) \in E$. We say that (v, w) is an edge *from v to w*, and that it is an *outgoing edge* of v and an *incoming edge* of w. We call v a *parent* of w and we call w a *child* of v. We say that the edge (v, w) is *incident* to v and w.

The *child set* of a vertex is the set of its child vertices and similarly for the *parent set*; we refer to these sets as the *children* and *parents* of the vertex, respectively. The *indegree* of a vertex is number parents it has and the *outdegree* is the number of children it has.

The *parents*, *children*, and *neighbors* of a *set A* of vertices each defined to be the set of vertices which *are not* in the set but *are* the parents, children or neighbors of some vertex in the set defined.

A vertex is a *source* vertex if it only has outgoing edges (i.e., is the child of no vertex its parent set is empty) and a vertex is a *sink* if it only has incoming edges (i.e., is the parent of no vertex).

A directed graph is *complete* if every vertex is both a child and parent of every other vertex.

Notation

We denote by $\mathrm{pa} : V \to \mathcal{P}(V)$ and $\mathrm{ch} : V \to \mathcal{P}(V)$ the functions associating to each vertex its set of parents and set of children, respectively. As usual, we denote the parents of vertex v by pa_v and the children by ch_v.

Self-loops

If x is a vertex, and (x, x) is an edge, we call such an edge a *self-loop* (or just *loop*). Many authorities exclude self-loops in their definition of directed graphs, but we allow them. To make the distinction, we call a graph with no *loops simple* (a *simple graph*).

Directed Graphs

Set Numbers

Finite Sets

Equivalent Sets

Natural Order　　　　　　Function Inverses

Function Composites　　　Function Images

Equivalence Relations　　Functions

Peano Axioms　　　　　　Relations

Natural Induction

Ordered Pair Projections

Set Products

Natural Numbers　　　　Set Powers

Intersection of Empty Set　　Successor Sets　　Unordered Triples

Pair Unions　　Set Partitions　　Ordered Pairs

Set Differences　　Set Intersections　　Set Unions

Pair Intersections　　Empty Set　　Unordered Pairs

Set Specification

Set Equality

Set Inclusion

Deductions

Quantified Statements

Logical Statements

Statements

Identities　　Sets

Names

Letters　　Objects

4