Discrete structures

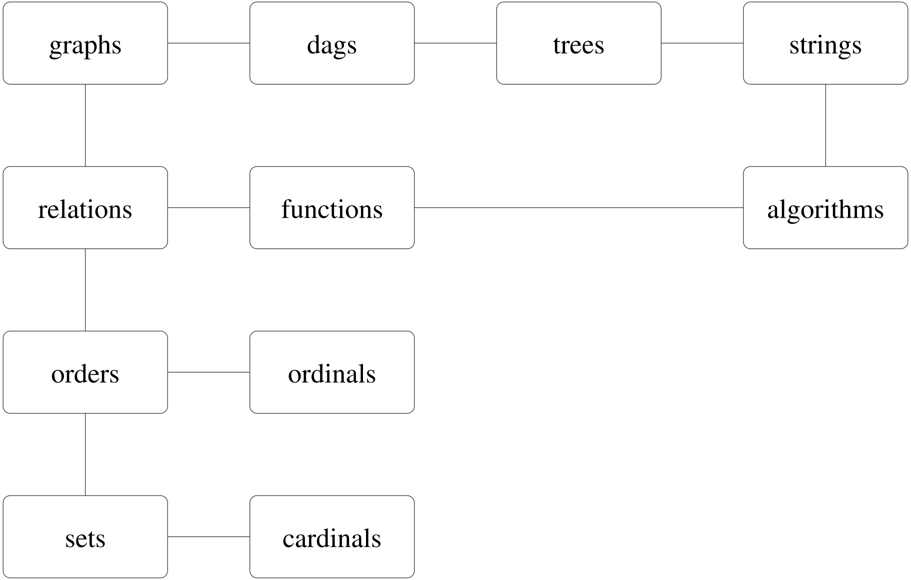
Vincent van Oostrom

http://cl-informatik.uibk.ac.at

Course themes

- directed and undirected graphs
- relations and functions
- orders and induction
- trees and dags
- finite and infinite counting
- elementary number theory
- Turing machines, algorithms, and complexity
- decidable and undecidable problem

Discrete structures



Questions and methodology

- When are two structures the same?
- When is one structure a substructure of another?
- How can we represent structures?
- What operations can we do on the structures?

Questions and methodology

- When are two structures the same?
- When is one structure a substructure of another?
- How can we represent structures?
- What operations can we do on the structures?
- Specify structures and operations mathematically
- Implement operations on structures by algorithms
- Prove that algorithm implement the operations, using appropriate mathematical techniques
- This course: basic discrete structures and basic mathematical techniques

Graphs for modelling problems

- History: Euler's seven bridges
- Sameness: Graph isomorphism problem
- Map drawing: Four color theorem
- Graph drawing: Kuratowski graph planarity
- Networks: Maximum flow problem
- Social networks: Friendship paradox
- flow graphs, abstract syntax trees, neural networks, ...

Definition (Directed multigraph)

A directed multigraph G is given by

- a set V of vertices or nodes
- a set E of edges
- functions $src\colon E \to V$ and $tgt\colon E \to V$ that map an edge e to its beginning or source $src(e)$ respectively end or target $tgt(e)$
- e is an edge from $src(e)$ to $tgt(e)$, its direction

Definition (Directed multigraph)

A directed multigraph G is given by

- a set V of vertices or nodes
- a set E of edges
- functions $src: E \to V$ and $tgt: E \to V$ that map an edge e to its beginning or source $src(e)$ respectively end or target $tgt(e)$
- e is an edge from $src(e)$ to $tgt(e)$, its direction

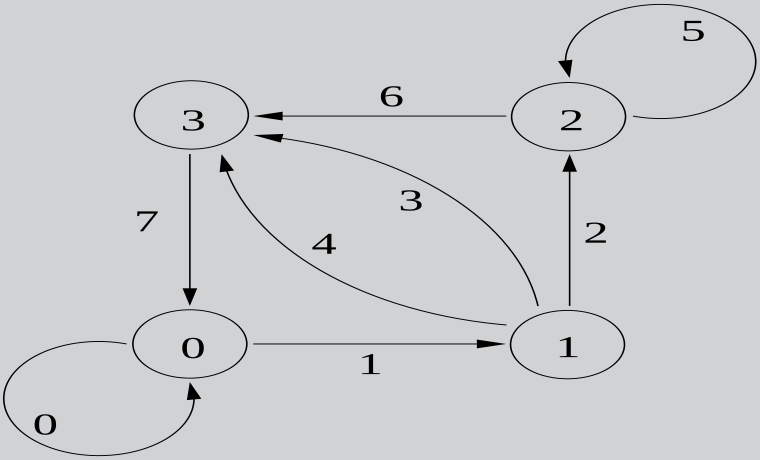
Example

Let $V = \{0, 1, 2, 3\}$, $E = \{0, 1, 2, \ldots, 7\}$ and the functions src and tgt be given by

e	$src(e)$	$tgt(e)$
0	0	0
1	0	1
2	1	2
3	1	3

e	$src(e)$	$tgt(e)$
4	1	3
5	2	2
6	2	3
7	3	0

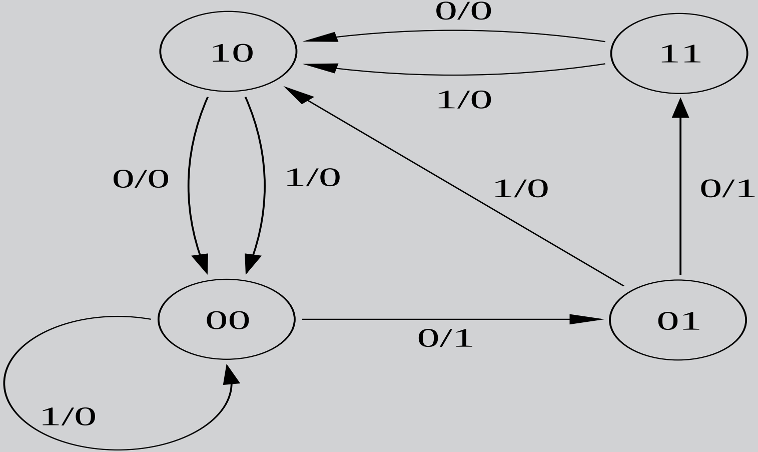
Example (Continued)



Definition

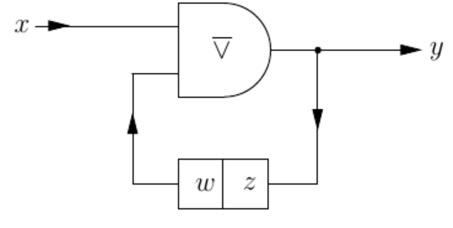
- Node *c* is an immediate predecessor of node *d*, if there is an edge from *c* to *d*
- *d* is then an immediate successor of *c*
- a loop is an edge from a node to itself
- edges having the same sources and the same edges are said to be parallel
- the number of edges having *e* as target is the indegree of *e*
- the number of edges having *e* as source is the outdegree of *e*
- a multigraph is called labelled if there are functions from the nodes or edges to some set of labels.
- if labels are numbers, then we speak of weighted graphs

Example



Example (Continued)

The previous graph is the state-diagram of a synchronous circuit with input x, output y, a NOR-gate and a buffer of length 2



the equations for the bit streams are

$$\begin{aligned} y(t) &= x(t) \,\overline{\vee}\, w(t) \\ w(t+1) &= z(t) \\ z(t+1) &= y(t) \end{aligned}$$

indexed by $t \in \mathbb{N}$ discrete time

Definition

- A directed graph (or digraph) is a directed multigraph without parallel edges
- for every pair (c, d) of nodes there is at most one edge e from c to d
- instead of the edge e we may write the pair (c, d)

Definition

- A directed graph (or digraph) is a directed multigraph without parallel edges
- for every pair (c, d) of nodes there is at most one edge e from c to d
- instead of the edge e we may write the pair (c, d)

Example

Let R be a relation on a set M. Then the digraph of R is given by:

- the set of nodes M
- the set of edges R
- the functions $src((x, y)) = x$ and $tgt((x, y)) = y$

Definition

- Let $G = (V, E, src, tgt)$ be a directed multigraph
- $G' = (V', E', src', tgt')$ is a sub-multigraph of G, if $V' \subseteq V$, $E' \subseteq E$ and $src'(e) = src(e)$, $tgt'(e) = tgt(e)$ for all $e \in E'$
- A sub-graph is a sub-multigraph that is a graph

Definition

- Let $G = (V, E, src, tgt)$ be a directed multigraph
- $G' = (V', E', src', tgt')$ is a sub-multigraph of G, if $V' \subseteq V$, $E' \subseteq E$ and $src'(e) = src(e)$, $tgt'(e) = tgt(e)$ for all $e \in E'$
- A sub-graph is a sub-multigraph that is a graph

Definition

Let (V, E, src, tgt) be a directed multigraph with nodes c, d

- A tuple $(e_0, e_1, \ldots, e_{\ell-1}) \in E^\ell$ is a path from c to d of length ℓ, if there are nodes $v_0, v_1, \ldots, v_\ell$ such that $v_0 = c$, $v_\ell = d$, with $src(e_i) = v_i$ and $tgt(e_i) = v_{i+1}$ for $i = 0, 1, \ldots, \ell - 1$
- v_0 is the source node
- v_ℓ is the target node
- $v_1, v_2, \ldots, v_{\ell-1}$ are the intermediate nodes

Definition (Continued)

- the empty tuple $() \in E^0$ is the empty path from any node e, with source, target e
- a multigraph is strongly connected if there is a path from each node to each node
- a path is simple if non-empty and has pairwise distinct nodes (exception $v_0 = v_\ell$)
- the composition of paths $(e_0, e_1, \ldots, e_{\ell-1})$ (from c to d) and $(f_0, f_1, \ldots, f_{m-1})$ (from d to e) is a path from c to e given by

$$(e_0, e_1, \ldots, e_{\ell-1}, f_0, f_1, \ldots, f_{m-1})$$

Definition (Continued)

- the empty tuple $() \in E^0$ is the empty path from any node e, with source, target e
- a multigraph is strongly connected if there is a path from each node to each node
- a path is simple if non-empty and has pairwise distinct nodes (exception $v_0 = v_\ell$)
- the composition of paths $(e_0, e_1, \ldots, e_{\ell-1})$ (from c to d) and $(f_0, f_1, \ldots, f_{m-1})$ (from d to e) is a path from c to e given by

$$(e_0, e_1, \ldots, e_{\ell-1}, f_0, f_1, \ldots, f_{m-1})$$

Definition

Let (V, E, src, tgt) be a directed multigraph having finitely many nodes- and edges; we number the nodes as $v_0, v_1, \ldots, v_{n-1}$. The matrix $A \in \mathbb{N}^{n \times n}$,

$$A_{ij} := \#(\{e \in E \mid src(e) = v_i \text{ and } tgt(e) = v_j\})$$

is the adjacency matrix

Example

The adjacency matrix for the multigraph of the first example is

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Shortest paths

Definition

Let G be a directed multigraph with non-negative edge-weight given by w

- The length or weight of a path $(e_0, e_1, \ldots, e_{\ell-1})$ with respect to w is the sum of the weights $w(e_i)$ of its edges e_i
- The distance from node e to node d is the minimal length of a path from e to d, if that exists, and ∞ otherwise

Algorithm of Floyd, distance initialisation

Definition

- Let G be a directed multigraph with finite sets of nodes V and edges E, and a non-negative edge-weights w
- We number the nodes $v_0, v_1, \ldots, v_{n-1}$
- Let B be the $n \times n$-matrix with elements

$$B_{ij} := \begin{cases} 0 & \text{if } i = j \\ \min\{w(e) \mid e \text{ edge from } v_i \text{ to } v_j\} & i \neq j \text{ and edge from } v_i \text{ to } v_j \text{ exists} \\ \infty & \text{otherwise} \end{cases}$$

Example

From adjacency matrix to distance matrix before Floyd

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} 0 & 1 & \infty & \infty \\ \infty & 0 & 1 & 1 \\ \infty & \infty & 0 & 1 \\ 1 & \infty & \infty & 0 \end{pmatrix}$$

Algorithm of Floyd

Theorem

The following algorithm overwrites the matrix B with the matrix of distances

> *For r from 0 to $n-1$ repeat:*
>
>   *Set $N = B$.*
>
>   *For i from 0 to $n-1$ repeat:*
>
>   *For j from 0 to $n-1$ repeat:*
>
>   *Set $N_{ij} = \min(B_{ij}, B_{ir} + B_{rj})$.*
>
>   *Set $B = N$.*

Example

Distances matrix after Floyd

$$\begin{pmatrix} 0 & 1 & 2 & 2 \\ 2 & 0 & 1 & 1 \\ 2 & 3 & 0 & 1 \\ 1 & 2 & 3 & 0 \end{pmatrix}$$

Properties of Floyd's algorithm

- Does it work? What does that mean, exactly?
- In what language do we express that?
- How do we prove it?
- Why does the algorithm work?
- How fast is it? As a function of what?
- How much memory does it use?
- How do we express this in a computer-independent way?
- . . .