Unit 47

- Matrix algorithms for analysis and synthesis of binary relations R on a set M
- Extensions and applications in the field of mathematics, computer science and algorithm engineering.

Lecture in the class given by Erwin P. Stoschek at the Chonqing University, Dept. Computer Science, recorded and illuminated by Liu Tianhai.

Part 1: Matrix algorithms for analysis and synthesis of binary relations R on a set M

Binary relation on a set

A binary relation R on a set M = {m_1, m_2, m_3,..., m_p}; card(M) = p
represents a defined relationship between two elements each of M, expressed by the

- set notation

  R = {(m_i, m_j) | m_i, $m_j \in$ M and m_i is related to m_j}

  $R \subseteq M \times M$

  $\underbrace{M \times M}$ = {($\underbrace{m_i, m_j}$) | m_i, $m_j \in$ M}

  ordered pair

  Cartesian product

- element notation

m_i is related to m_j by R		m_i is not related to m_j by R
(m_i, m_j) $\in$ R	postfix	(m_i, m_j) $\notin$ R
$m_i R m_j$; xRy	infix	$\overline{m_i R m_j} = m_i \overline{R} m_j$
R(m_i, m_j)	prefix	
	notation	

.

Examples:

- $x \geqq y$; $x < y$; $x \mid y$; $x \equiv y \pmod{k}$ on N
- word w_i is containing in word w_j on A^*, f.e.
  UGACU $\subset$ GUGAUGACUUAACG
- m_i is successor of m_j in a sequence or in a operation
- component m_i is built in component m_j
- pin m_i of an integrated circuit is connected with pin m_j
- species m_i is ancestor of species m_j

Representation of relations R on M

- Set notation, listing all elements of R

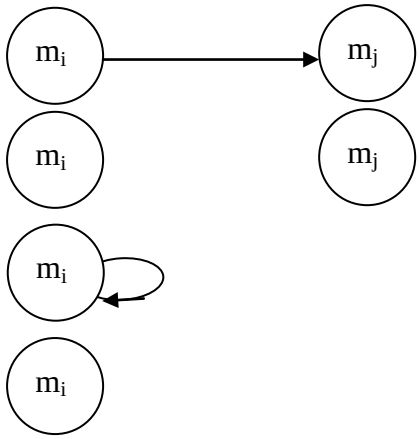
  R = {(a, a), (a, b), (b, a), (a, c)} $\subset$ {a, b, c} $\times$ {a, b, c}
- Set notation, using set builders

  R = {(x, y) | x, y $\in$ {1, 2, 3, 4, 5, 6, 7, 8} and x | y}

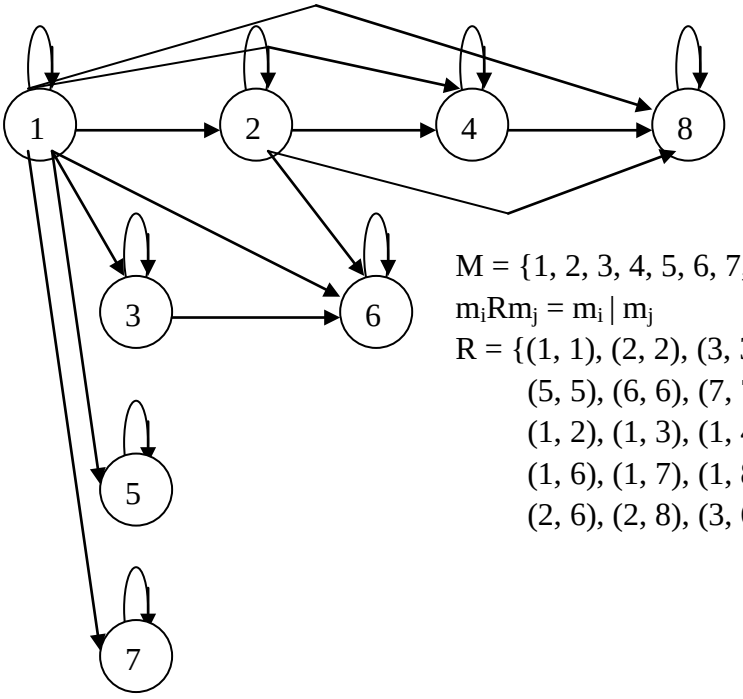
  x | y – x divides y (without remainder) ;

  $\text{int}(\frac{y}{x}) = div(y, x) = \frac{y}{x}$, y mod x = rest(y, x) = 0

- Element notation, using predicates xRy = “x | y”
- Relation graph:
  directed graph with set of vertices M and set of edges R

directed edges of the relation graph	relation
	$m_i R m_j$ $\overline{m_i R m_j} = m_i \overline{R} m_j$ $m_i R m_i$ $m_i \overline{R} m_i$

Example



M = {1, 2, 3, 4, 5, 6, 7, 8}
$m_i R m_j = m_i \mid m_j$
R = {(1, 1), (2, 2), (3, 3), (4, 4),
(5, 5), (6, 6), (7, 7), (8, 8),
(1, 2), (1, 3), (1, 4), (1, 5),
(1, 6), (1, 7), (1, 8), (2, 4),
(2, 6), (2, 8), (3, 6), (4, 8)}

- Characteristic function of the relation R:
  unique mapping f: $M \times M \rightarrow \{0, 1\}$ from $M \times M$ to {0, 1}

$f(m_i, m_j) = f[i, j]$	if
1	$m_i R m_j$

0	$m_i \overline{R} m_j$

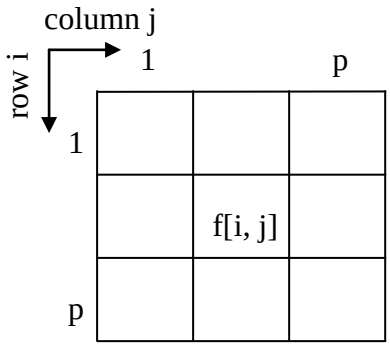
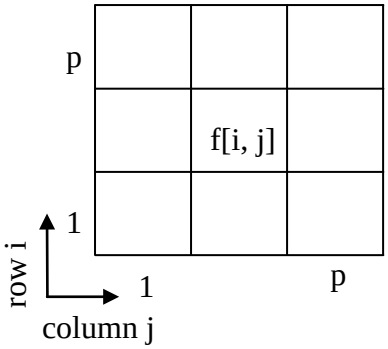
Relation matrix

$p \times p$ (p by p) matrix

= value table of $f(m_i, m_j)$

= adjacency matrix of the relation graph

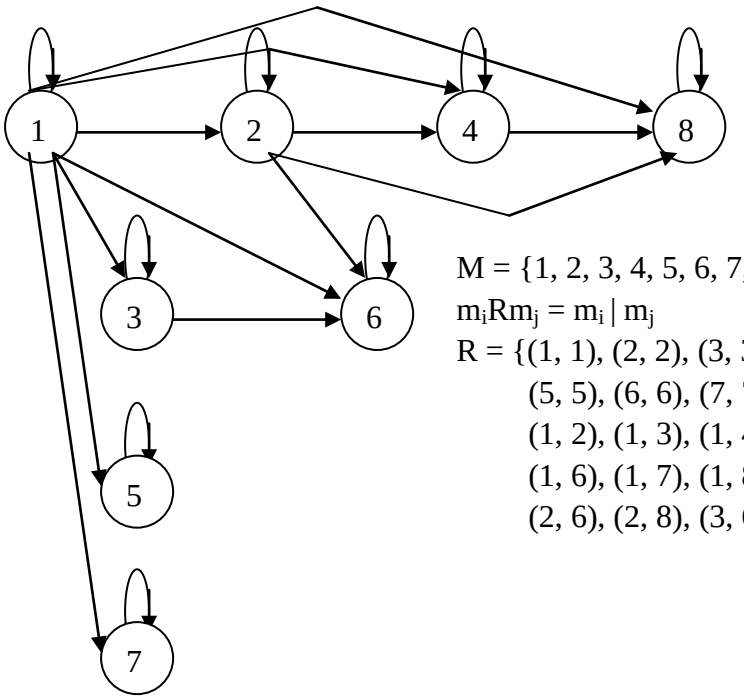
$\underline{f} = (f(m_i, m_j))_{i=1,\, j=1}^{p \quad p} = (f[i, j])_{i=1,\, j=1}^{p \quad p}$,

classical notation	Cartesian notation
(mathematics, physics,...)	(pattern recognition, image processing,...)
	
→ classical matrix operations	→ algorithmic matrix operations!

The representation of a binary relation R on M by its relation matrix is the most frequently used (especially for large p) basis for the algorithmic and computer handling (analysis and synthesis) of this relation in many applications.

Example: “divides” relation

Relation graph

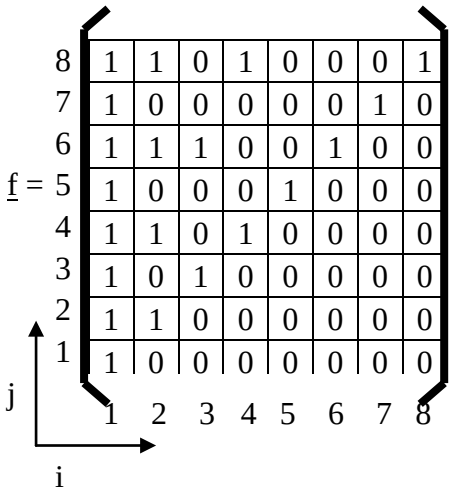


M = {1, 2, 3, 4, 5, 6, 7, 8}

$m_i R m_j = m_i \mid m_j$

R = {(1, 1), (2, 2), (3, 3), (4, 4),
(5, 5), (6, 6), (7, 7), (8, 8),
(1, 2), (1, 3), (1, 4), (1, 5),
(1, 6), (1, 7), (1, 8), (2, 4),
(2, 6), (2, 8), (3, 6), (4, 8)}

and relation matrix (Cartesian notation)



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

The following special binary relations are very important.

A binary relation $R \subseteq M \times M$ is called ..,	if the condition .. is true R-notation	f-notation
reflexive	$(\forall x)$ xRx $x \in M$	$(\forall x)$ f(x, x) = 1 $x \in M$
irreflexive	$(\forall x)x\overline{R}x$	$(\forall x)$ f(x, x) = 0
symmetric	$(\forall x)(\forall y)xRy \leftrightarrow yRx$	$(\forall x)(\forall y)$ f(x, y) = f(y, x)
antisymmetric	$(\forall x)(\forall y)xRy \rightarrow y\overline{R}x$ $x \neq y$	$(\forall x)(\forall y)\overline{f(x,y)} \vee \overline{f(y,x)} = \overline{f(x,y) \wedge f(y,x)} = 1 \Rightarrow$ $x \neq y$ $(\forall x)(\forall y)\, f(x,y) \wedge f(y,x) = 0$ $x \neq y$
asymmetric	$(\forall x)(\forall y)xRy \rightarrow y\overline{R}x$	$(\forall x)(\forall y)\, f(x,y) \wedge f(y,x) = 0$ From this follows $(\forall x)\, f(x,x) = 0$ i.g. an asymmetric binary relation is at the same time also irreflexive.

<table>
<tr><td>A binary relation $R \subseteq M \times M$ is called ..,</td><td>if the condition .. is true
R-notation</td><td>f-notation</td></tr>
<tr><td>transitive</td><td>$(\forall x)(\forall y)(\forall z) xRy \wedge yRz \rightarrow xRz$</td><td>$(\forall x)(\forall y)(\forall z)\, \overline{f(x,y) \wedge f(y,z)} \vee f(x,z) = 1$
This definition has the implicit form
$\overline{a} \vee b = 1$, expression = value.
It can be transformed into the explicit form
$b = a \vee b$, variable=expression, proof by
<table><tr><th>a</th><th>b</th><th>$\overline{a} \vee b$</th><th>$a \vee b$</th></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>see also: $x^2 = 2 \rightarrow x = \frac{2}{x}$ (Module 1)
With this we obtain finally an explicit formulated definition
$(\forall x)(\forall y)(\forall z)\, f(x,z) = f(x,z) \vee f(x,y) \wedge f(y,z)$</td></tr>
<tr><td>intransitive</td><td>$(\forall x)(\forall y)(\forall z) xRy \wedge yRz \rightarrow x\overline{R}z$</td><td>$(\forall x)(\forall y)(\forall z)\, \overline{f(x,y) \wedge f(y,z)} \vee \overline{f(x,z)} = 1 \rightarrow$
$(\forall x)(\forall y)(\forall z)\, f(x,z) = f(x,z) \dot{-} f(x,y) \wedge f(y,z)$
with $a \dot{-} b = a \wedge \overline{b}$ (inhibition, logical subtraction).
From this follows $(\forall x) f(x,x) = 0$, i.g. an intransitive binary relation is at the same time also irreflexive.</td></tr>
</table>

Table1 : Special binary relations on a set

		disjunction logical addition	conjunction logical multiplication	implication	equivalence	antivalence	inhibition logical subtraction
		a or b	a and b	if a then b a implin b	a if and only if b a iff b	a exor b a xor b	a minus b a inhibit b
		$a \vee b$	$a \wedge b$	$a \rightarrow b =$ $\bar{a} \vee b =$ $\begin{cases} 1 & a = b \\ \text{iff} & \\ b & a \neq b \end{cases}$	$a \leftrightarrow b =$ $(a \vee \bar{b}) \wedge (\bar{a} \vee b) =$ $\begin{cases} 1 & a = b \\ \text{iff} & \\ 0 & a \neq b \end{cases}$	$a \oplus b =$ $(a \wedge \bar{b}) \vee (\bar{a} \wedge b) =$ $\begin{cases} 0 & a = b \\ \text{iff} & \\ 1 & a \neq b \end{cases}$	$a \dot{-} b =$ $a \wedge \bar{b} =$ $\begin{cases} 0 & a = b \\ \text{iff} & \\ a & a \neq b \end{cases}$
a	b						
0	0	0	0	1	1	0	0
0	1	1	0	1	0	1	0
1	0	1	0	0	0	1	1
1	1	1	1	1	1	0	0
		$a \vee b = c$ a and b have equal access right on c	$a \wedge b = c$ a and b have only together access on c				

	negation
	not a
a	$\neg a, \bar{a}$
0	1
1	0

Transformation of implicit logical formulae into the form
"variable = expression" (→ "variable: = expression"):

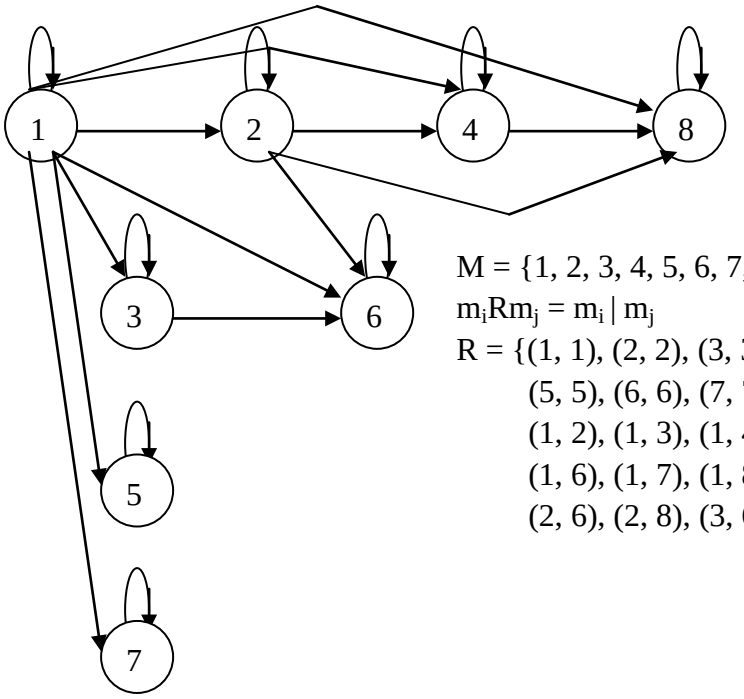
implicit formula expression = value	corresponding variable = expression
$a \vee b = 0$	$a = \bar{a} \wedge b$ $b = a \wedge \bar{b}$
$a \vee b = 1$	$a = a \vee \bar{b} = \overline{\bar{a} \wedge b}$ $b = \bar{a} \vee b = \overline{a \wedge \bar{b}}$
$a \wedge b = 0$	$a = a \wedge \bar{b}$ $b = \bar{a} \wedge b$
$a \wedge b = 1$	$a = \bar{a} \vee b = \overline{a \wedge \bar{b}}$ $b = a \vee \bar{b} = \overline{\bar{a} \wedge b}$
$a \oplus b = 0$	$a = b$
$a \oplus b = 1$	$a = \bar{b}$ $b = \bar{a}$
$a \oplus b = c$	$a = b \oplus c$ $b = a \oplus c$

Proofs by using truth tables.

A binary relation $R \subseteq M \times M$ is called ..,	if the condition .. is true	f-notation
equivalence relation	reflexivity symmetry transitivity	$(\forall x) f(x,x) = 1$ $(\forall x)(\forall y) f(x,y) = f(y,x)$ $(\forall x)(\forall y)(\forall z) f(x,z) = f(x,z) \vee f(x,y) \wedge f(y,z)$
irreflexive partial ordering	asymmetry transitivity	$(\forall x)(\forall y) f(x,y) \wedge f(y,x) = 0$ $(\forall x)(\forall y)(\forall z) f(x,z) = f(x,z) \vee f(x,y) \wedge f(y,z)$
reflexive partial ordering	reflexivity antisymmetry transitivity	$(\forall x) f(x,x) = 1$ $(\forall x)(\forall y) f(x,y) \wedge f(y,x) = 0 \quad x \neq y$ $(\forall x)(\forall y)(\forall z) f(x,z) = f(x,z) \vee f(x,y) \wedge f(y,z)$

Table 2: Special binary relations on a set

Example: “divides” relation



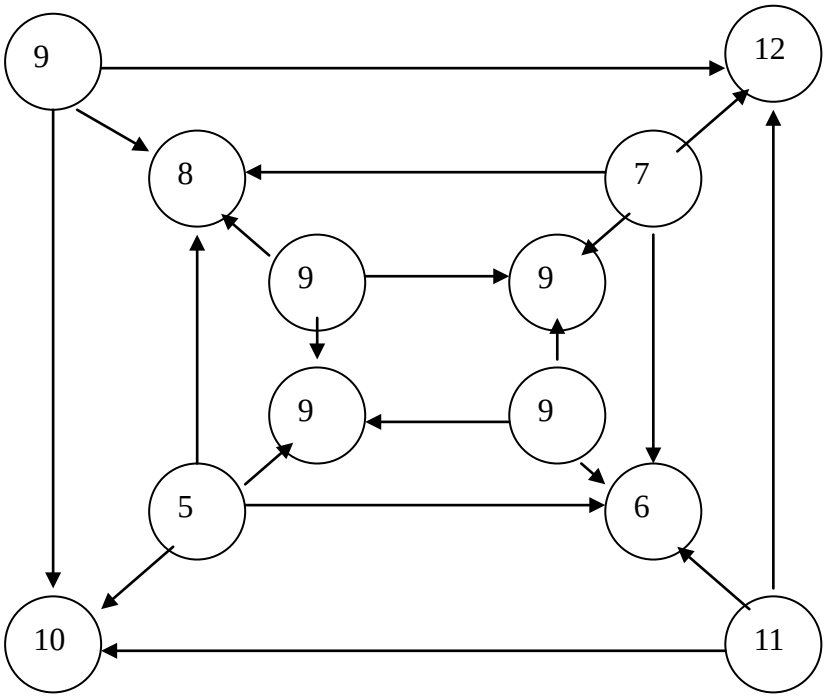
M = {1, 2, 3, 4, 5, 6, 7, 8}

$m_i R m_j = m_i \mid m_j$

R = {(1, 1), (2, 2), (3, 3), (4, 4),
(5, 5), (6, 6), (7, 7), (8, 8),
(1, 2), (1, 3), (1, 4), (1, 5),
(1, 6), (1, 7), (1, 8), (2, 4),
(2, 6), (2, 8), (3, 6), (4, 8)}

R is reflexive
antisymmetric } a reflexive partial ordering.
transitive

Question: is the given R on M = {1, 2, .., 12} irreflexive, asymmetric, transitive, intransitive?



j \ i	1	2	3	4	5	6	7	8	9	10	11	12
12							1		1		1	
11												
10					1				1		1	
9												
8			1		1		1		1			
7												
6	1				1		1				1	
5												
4	1		1		1							
3												
2	1		1				1					
1												

Implantation of the f-noted definitions of the special binary relations reflexive, irreflexive, symmetric, antisymmetric, and asymmetric into the mathematical model relation matrix, Cartesian notation, using the matrix operations

- Extraction of the main diagonal elements of the relation matrix $\underline{f}$

$$re(\underline{f}) = \begin{pmatrix} 0 & for & i \neq j \\ f[i,j] & & i = j \end{pmatrix} \begin{matrix} p \\ i=1, \end{matrix} \begin{matrix} p \\ j = 1, \end{matrix}$$

- Extraction of the symmetric part of the relation matrix $\underline{f}$

$$sym(\underline{f}) = \begin{pmatrix} 0 & for & f[i,j] \neq f[j,i] \\ f[i,j] & & f[i,j] = f[j,i] \end{pmatrix} \begin{matrix} p \\ i=1, \end{matrix} \begin{matrix} p \\ j = 1 \end{matrix},$$

gives the following $\underline{f}$ definition

A binary relation $R \subseteq M \times M$ is called ..,	if the condition .. is true, $\underline{f}$-Notation
reflexive	$re(\underline{f}) = \underline{e}$ (identity matrix)
irreflexive	$re(\underline{f}) = \underline{0}$ (zero matrix)
symmetric	$sym(\underline{f}) = \underline{f}$
antisymmetric	$sym(\underline{f}) = re(\underline{f})$
asymmetric	$sym(\underline{f}) = \underline{0}$

Table 3 : Special binary relations on a set, matrix definitions

- Now: implantation of transitivity and intransitivity definitions into $\underline{f}$

$$(\forall x)(\forall y)(\forall z)\, f(x,z) = f(x,z) \boxed{\underline{\vee}} f(x,y) \wedge f(y,z)$$
$$x, y, z \in M$$

$\Downarrow$ indexed notation: $f(x,y) \rightarrow f(m_i, m_j) \rightarrow f[i,k]$

$$(\forall i)(\forall j)(\forall k)\, f[i,k] = f[i,k] \boxed{\underline{\vee}} f[i,j] \wedge f[j,k]$$
$$i, j, k \in N[1, p]$$

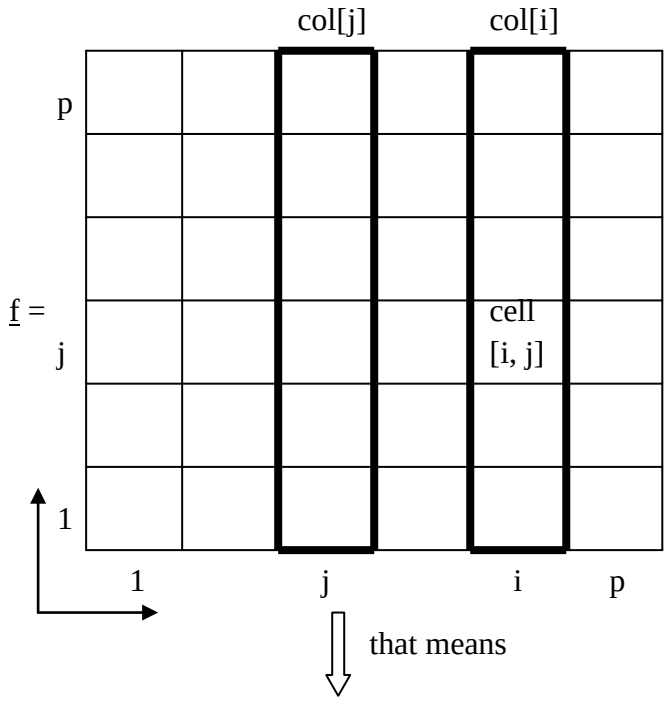
$\Downarrow$ separation of quantifications

$$(\forall i)(\forall j)((\underbrace{\forall k)\, f[i,k]} = f[i,k] \boxed{\underline{\vee}} f[i,j] \wedge f[j,k])$$

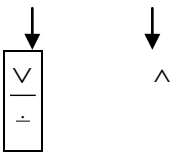
$\Downarrow$ implantation into $\underline{f}$

$$(\forall i)(\forall j)(column[i] = column[i] \boxed{\underline{\vee}} < i, j > \wedge column[j])$$

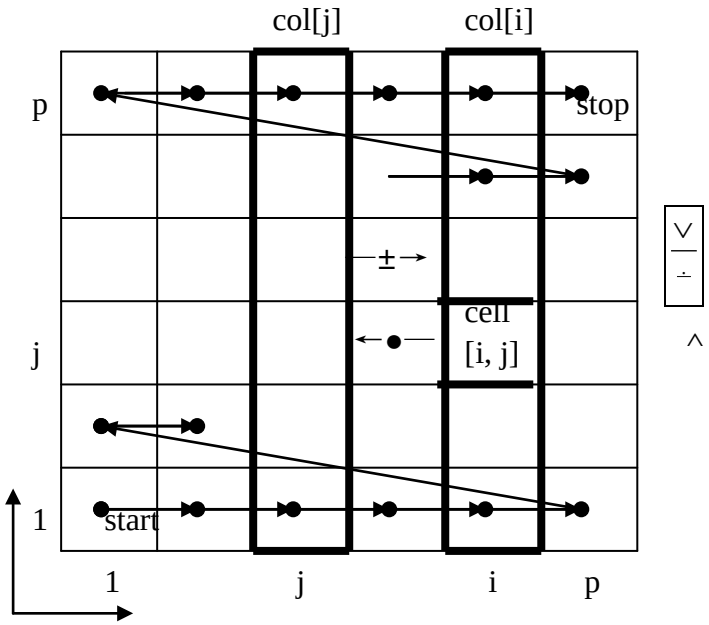
content of cell[i,j] = f[i,j]



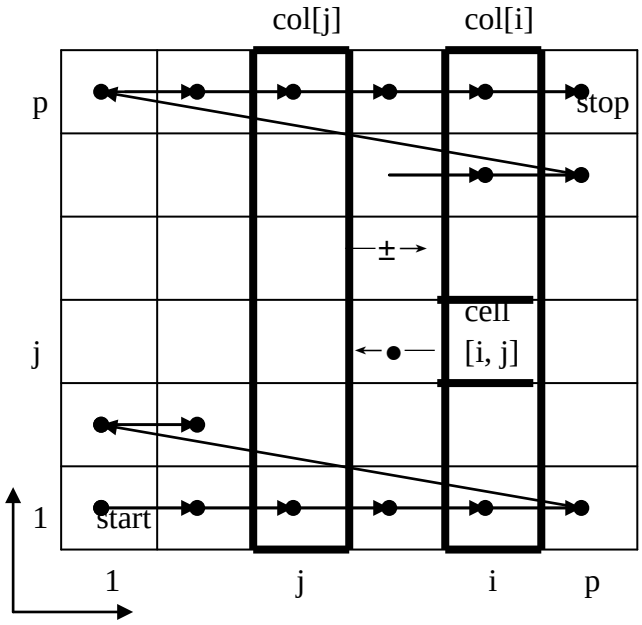
check for all cells of the matrix f the condition $col[i] = col[i] \pm \ < i, j > \bullet col[\, j]$ (triple operator),



for instance in the following sequential navigation order (line serially with flyback).

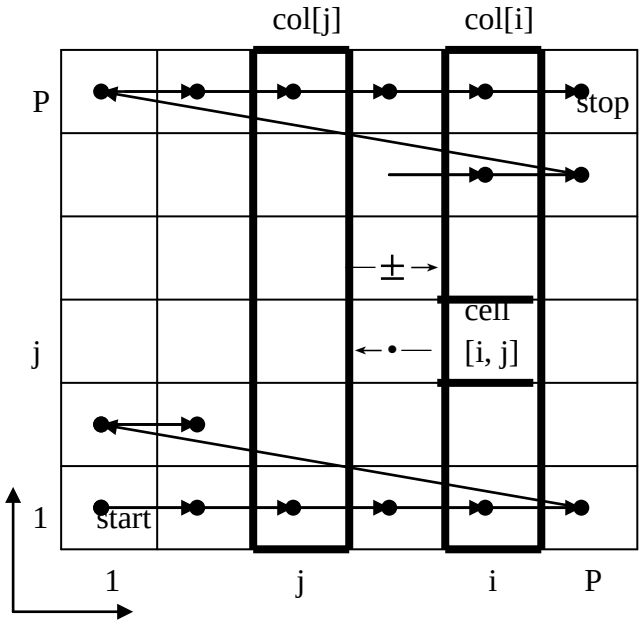


By extension "=" → ":=" we obtain the matrix operation "upwards iteration"



Input : p ; f
matrix : = f
j : = 1(1)p
i : = 1(1)p
yes f[i, j] ≠ 0 no
col[i] : = col[i]∨(f[i,j]∧)col[j]
it↓(f) : = matrix
Output : it (f) – upwards iteration

and downwards iteration



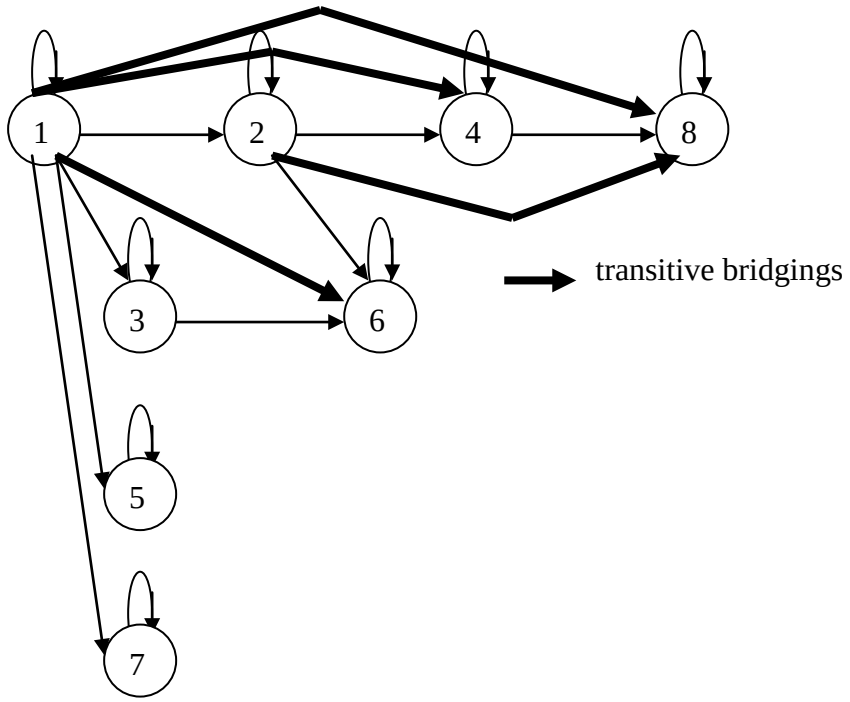
Input : p ; f
matrix : = f ÷ re(f)
j : = 1(1)p
i : = 1(1)p
yes f[i, j] ≠ 0 no
col[i] : = col[i] ÷ (f[i,j] ∧) col[j]
it ↑ (f) : = matrix ∨ re(f)
Output : it ↓ (f) – downwards iteration

and with that the f definitions

A binary relation $R \subseteq M \times M$ iscalled ..,	if the condition .. is true, f-notation
transitive intransitive	it ↑ (f) = f it ↓ (f) = f; re(f) = 0
equivalence relation	re(f) = e sym(f) = f it ↑ (f) = f
irreflexive partial ordering	re(f) = sym(f) = 0 it ↑ (f) = f
reflexive partial ordering	re(f) = sym(f) = e it ↑ (f) = f

Table 4 : Special binary relations on a set, matrix definitions.

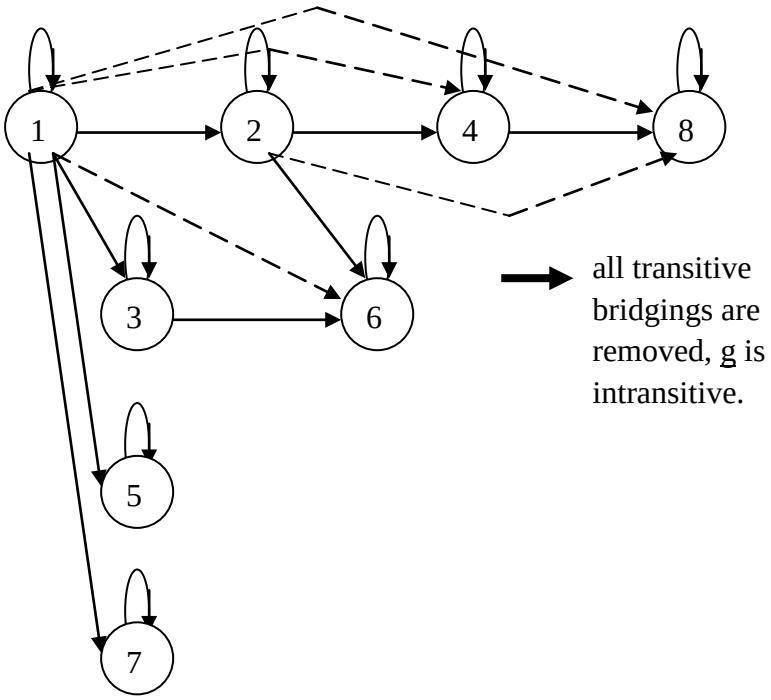
Example: “divides” relation (x|y)
(reflexive partial ordering)



$$\underline{f} = \begin{pmatrix} \underline{1} & \underline{1} & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & \underline{0} & 0 & 0 & 0 & 0 & 1 & 0 \\ \underline{1} & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & \underline{0} & 1 & 0 & 0 & 0 \\ \underline{1} & 1 & 0 & 1 & \underline{0} & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & \underline{0} & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & \underline{0} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & \underline{0} \end{pmatrix} = it \uparrow (\underline{f}) \quad : \underline{f} \text{ is transitive}$$

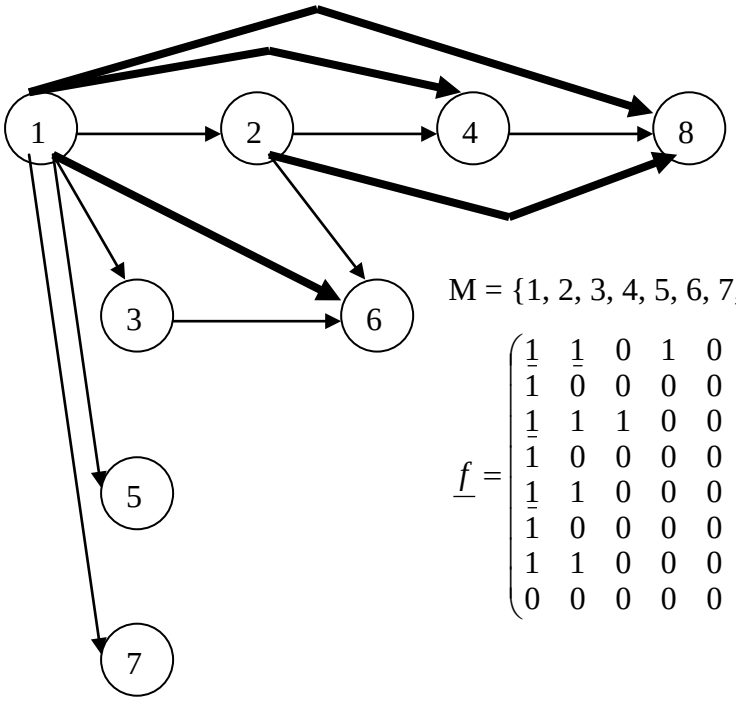
downwardsiteration

$$\underline{g} = it \downarrow (\underline{f}) = \begin{pmatrix} \underline{0} & \underline{0} & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & \underline{0} & 0 & 0 & 0 & 0 & 1 & 0 \\ \underline{0} & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & \underline{0} & 1 & 0 & 0 & 0 \\ \underline{0} & 1 & 0 & 1 & \underline{0} & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & \underline{0} & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & \underline{0} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & \underline{0} \end{pmatrix} = it \downarrow (\underline{g})$$



g (Hasse diagram) of the "divides" relation is its storage minimum!
From this storage minimum you can reconstruct f completely:
f = it ↑ (g).

Error in R = {(x, y)| x, y∈M and x|y and x≠y}



M = {1, 2, 3, 4, 5, 6, 7, 8}

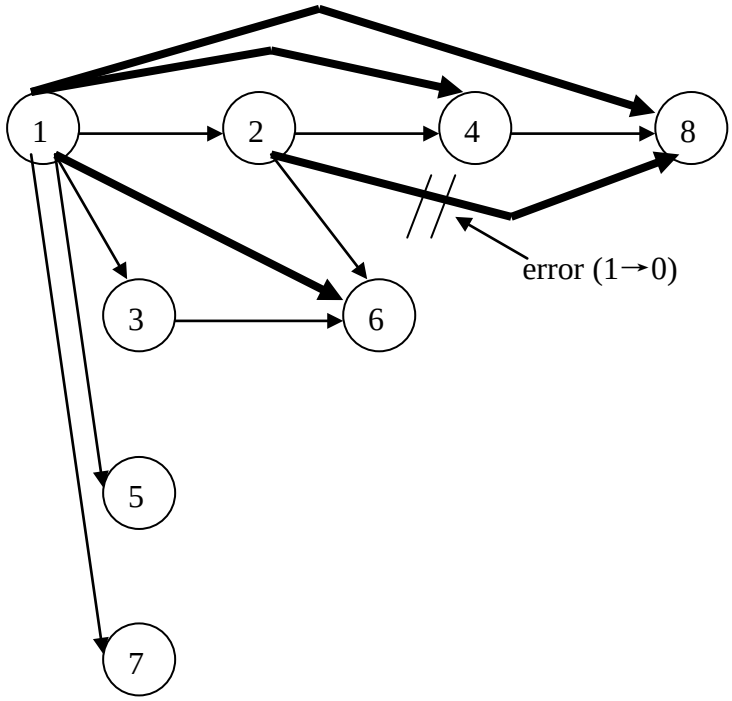
$$\underline{f} = \begin{pmatrix} \underline{1} & \underline{1} & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = it \uparrow (\underline{f})$$

= removal of an ordered pair (x, y), x≠y from R
= removal of the directed edge x→y from relation graph of R
= falsification of f(x, y) =1 into f'(x, y) = 0 in the relation matrix of R.

Error correction in cell(x, y) of $\underline{f}'$ by upwards iteration

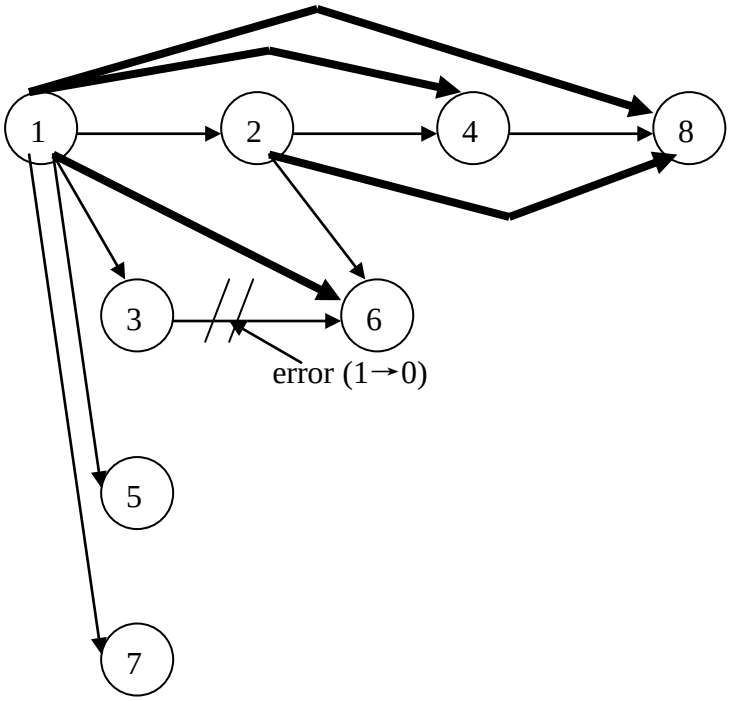
if and only if $(\underset{z\in M}{\exists z})\, f'(x,z) \wedge f'(z,y) = 1$

= reconstruction of the removed transitive bridging



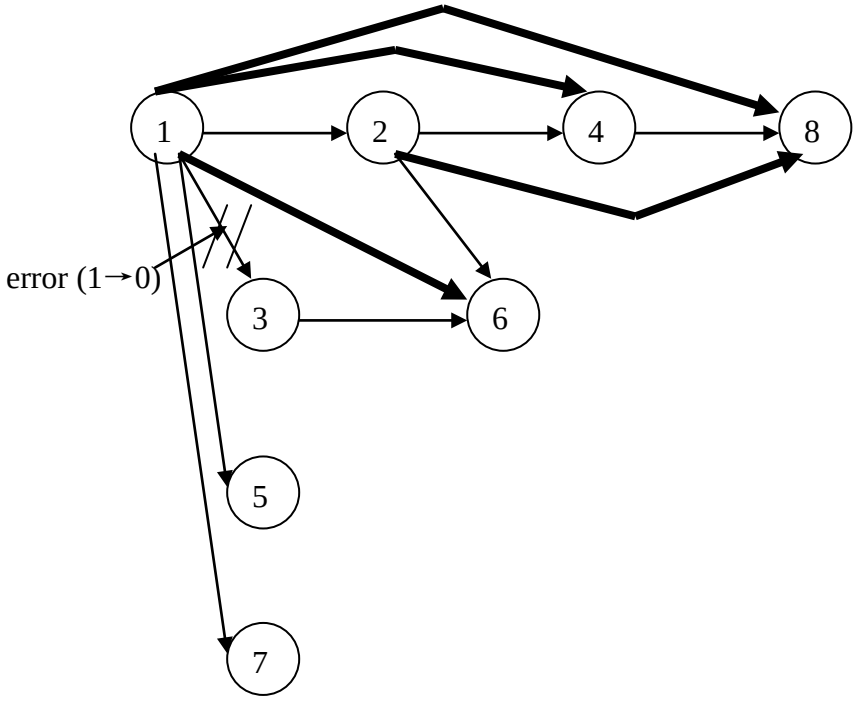
$$\underline{f}' = \begin{pmatrix} \underline{1} & \underline{\underline{0}} & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$it\uparrow(\underline{f}') = \begin{pmatrix} \underline{1} & \underline{1} & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \underline{f}$$: successful correction



$$\underline{f}' = \begin{pmatrix} \underline{1} & \underline{1} & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & \underline{0} & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & \underline{\underline{0}} & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

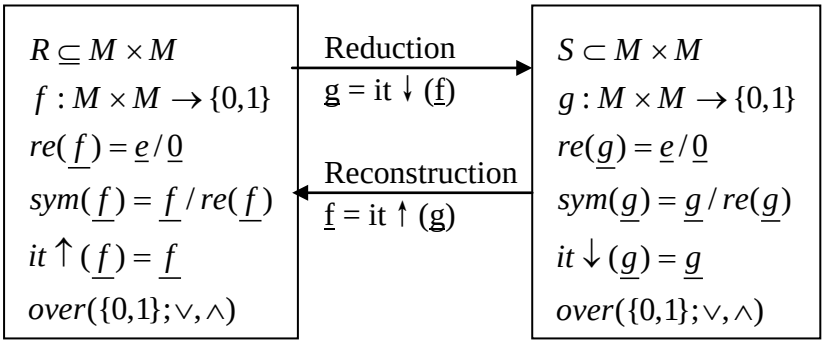
$$it\uparrow(\underline{f}') = \begin{pmatrix} \underline{1} & \underline{1} & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & \underline{0} & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & \underline{\underline{0}} & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \underline{f}' \neq \underline{f} \quad : \text{correction impossible}$$



$$\underline{f}' = \begin{pmatrix} \underline{1} & \underline{1} & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{\underline{1}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$it \uparrow (\underline{f}') = \begin{pmatrix} \underline{1} & \underline{1} & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{0} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \underline{\underline{1}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \underline{f}' \neq \underline{f} \quad : \text{correction impossible}$$

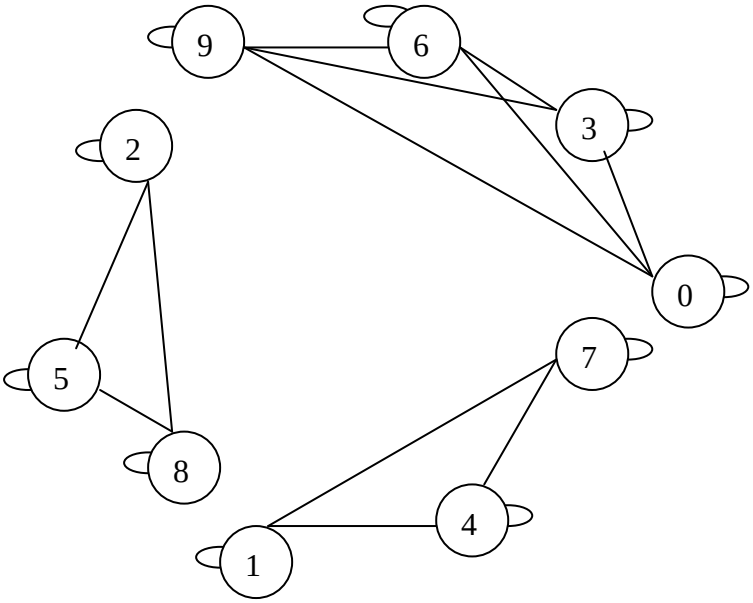
The preceding examples illustrates the inversion theorem of transitive relations:



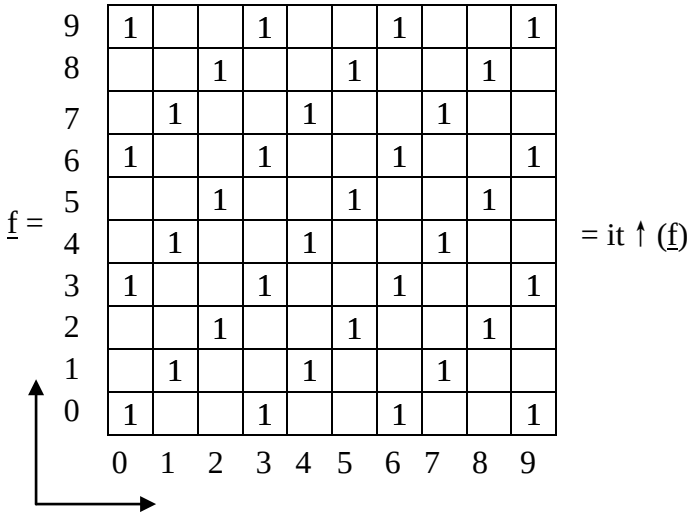
- Reduction: downwards iteration of the relation matrix $\underline{f} \Rightarrow \underline{g}$, removing of all transitive bridgings in the relation graph $\Rightarrow$ skeleton = storage minimum of R
- Reconstruction: upwards iteration of the skeleton matrix $\underline{g} \Rightarrow \underline{f}$, adding of all transitive bridgings to the skeleton = algorithmic (sequential, dynamic) reconstruction of the original relation from its storage minimum (genome)

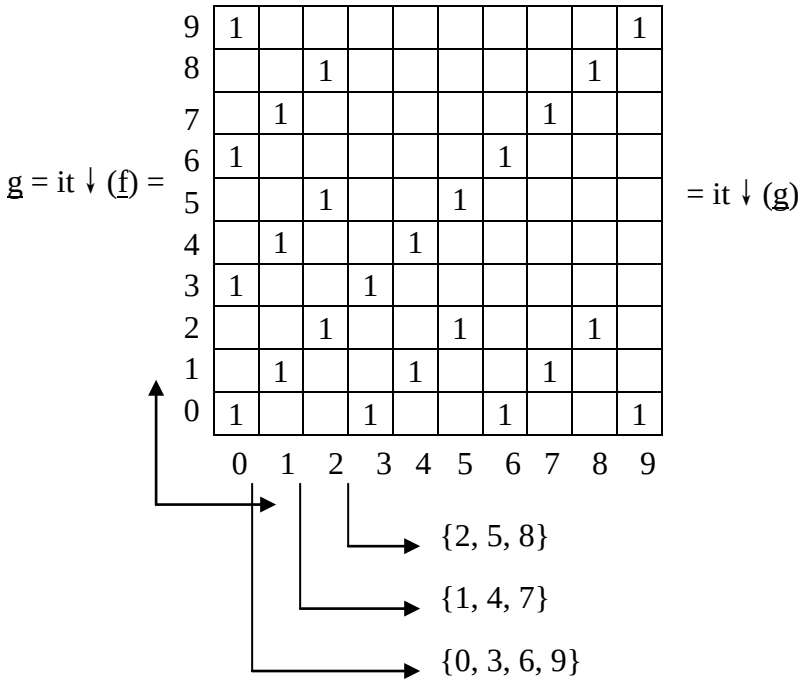
Example: congruence relation $x \equiv y \pmod m$ (equivalence relation)

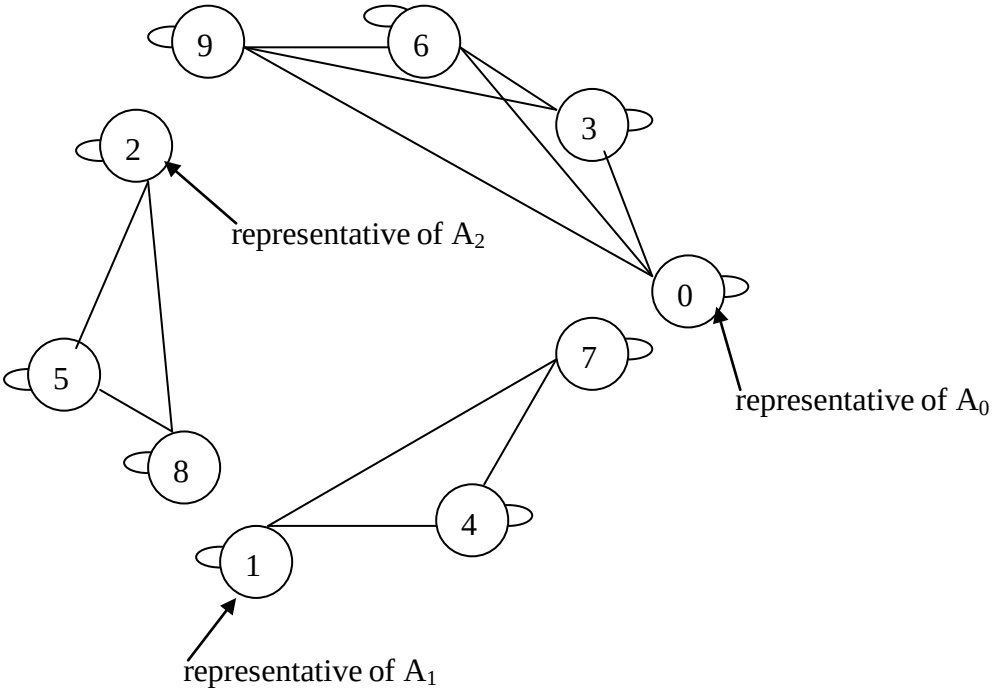
$$x \equiv y(\mathrm{mod}\, m) \Leftrightarrow m \mid (x - y);\ m \in \mathrm{N};\quad x, y \in \mathrm{N}_0$$
$$\overline{x \equiv y(\mathrm{mod}\, m)} \Leftrightarrow \overline{m \mid (x - y)}$$



M = N[0, 9]
xRy = "$x \equiv y \pmod 3$"


f =
9 1 1 1 1
8 1 1 1
7 1 1 1
6 1 1 1 1
5 1 1 1
4 1 1 1
3 1 1 1 1
2 1 1 1
1 1 1 1
0 1 1 1 1
0 1 2 3 4 5 6 7 8 9
= it ↑ (f)


g = it ↓ (f) =
9 1 1
8 1 1
7 1 1
6 1 1
5 1 1 1
4 1 1 1
3 1 1 1
2 1 1 1 1
1 1 1 1 1
0 1 1 1 1
0 1 2 3 4 5 6 7 8 9
{2, 5, 8}
{1, 4, 7}
{0, 3, 6, 9}
= it ↓ (g)



The equivalence relation R splits M into equivalence classes A_i
M = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

$$\overset{(R)}{=} \underbrace{\{0,3,6,9\}}_{=A_0} \cup \underbrace{\{1,4,7\}}_{=A_1} \cup \underbrace{\{2,5,8\}}_{=A_2}$$

with

- $(\forall i) A_i \neq \Phi$ nonempty
- $(\forall i)(\forall j) A_i \cap A_j = \Phi$ disjunct
- $A_1 \cup A_2 \cup A_3 \cup ... \cup A_m = M$ complete
- $(\forall i)(\forall x)(\forall y) xRy$
  $x, y \in A_i$
- $(\forall i)(\forall j)$ $(\forall x)$ $(\forall y)$ $x\overline{R}x$.
  $i \neq j$ $x \in A_i$ $y \in A_j$

Adjacency matrix and connectivity matrix of a directed graph with p vertices

M = {m_1, m_2, m_3,..., m_p}

$\underline{f} = \left(f(m_i, m_j)\right)_{i=1,\,j=1}^{p\quad p}$ - adjacency matrix

$f(m_i, m_j) = 1$ - there is a directed edge from m_i to m_j

$f(m_i, m_j) = 0$ - there isn't a directed edge from m_i to m_j

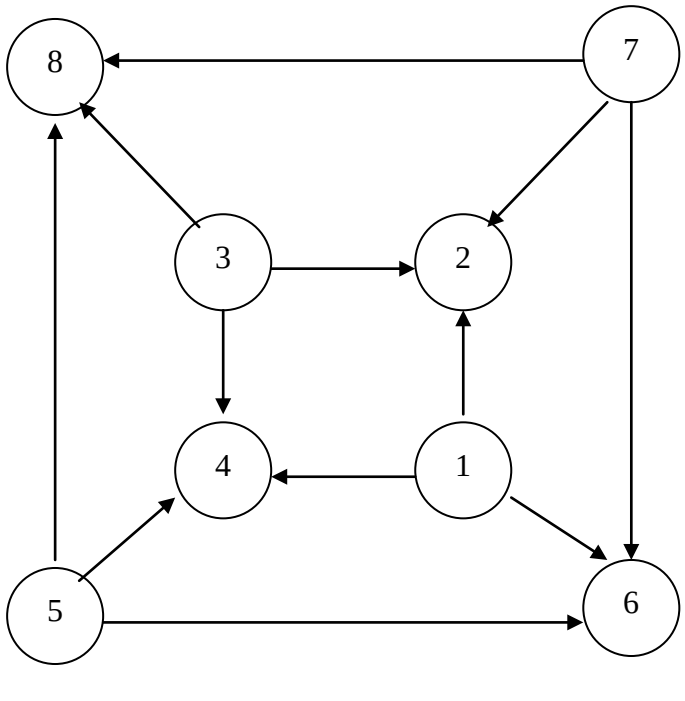
$\underline{c} = \left(c(m_i, m_j)\right)_{i=1,\,j=1}^{p\quad p} = it \uparrow (\underline{f})$ - connectivity matrix (transitive closure of $\underline{f}$)

$c(m_i, m_j) = 1$ - there is a directed path (sequence of successive edges) from m_i to m_j

$c(m_i, m_j) = 0$ - there isn´t a directed path from m_i to m_j

Loop finder in an asymmetric binary relation R in M
R is loop free if and only if re(f) = sym(f) = re(it ↑ (f)) = 0
Example 1:

re(f) = sym(f) = re(it ↑ (f)) = 0 ⇒
R is loop free (not any closed path in the relation graph)



M = N[1, 8],
relation graph:

f =

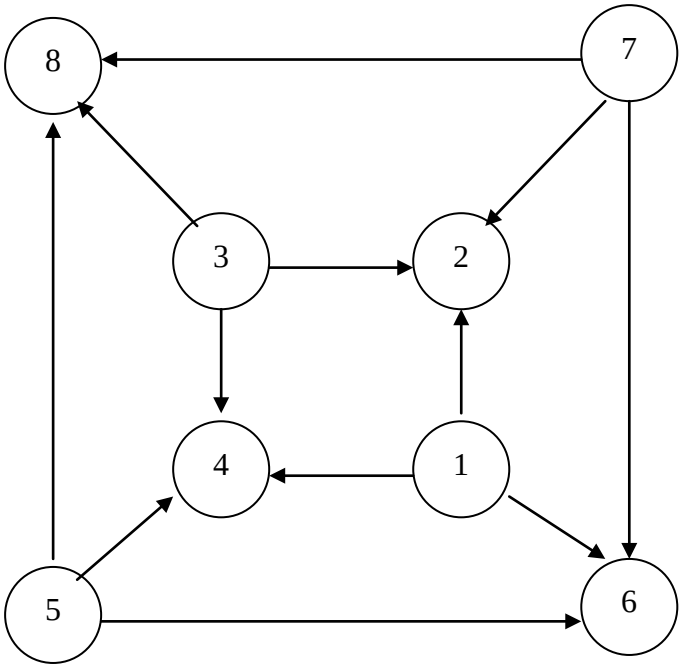
0	0	1	0	1	0	1	0
0	0	0	0	0	0	0	0
1	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0
1	0	1	0	1	0	0	0
0	0	0	0	0	0	0	0
1	0	1	0	0	0	1	0
0	0	0	0	0	0	0	0

8 7 6 5 4 3 2 1

1 2 3 4 5 6 7 8

= it ↑ (f)

Example 2

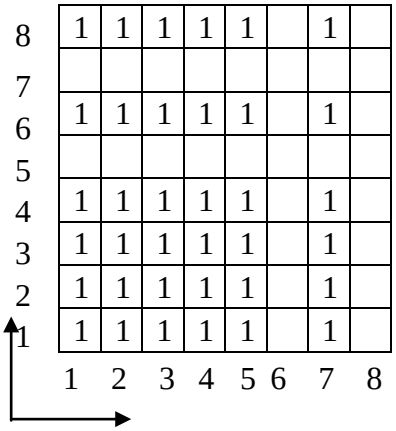


M = N[1, 8],
relation graph:

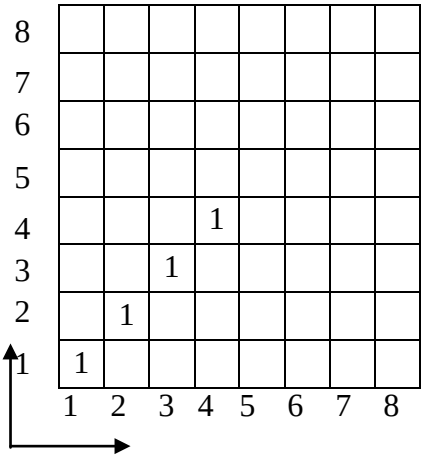
f =

8			1		1		1	
7								
6	1				1		1	
5								
4	1				1			
3				1				
2			1					1
1		1						
	1	2	3	4	5	6	7	8

it ↑ (f) =



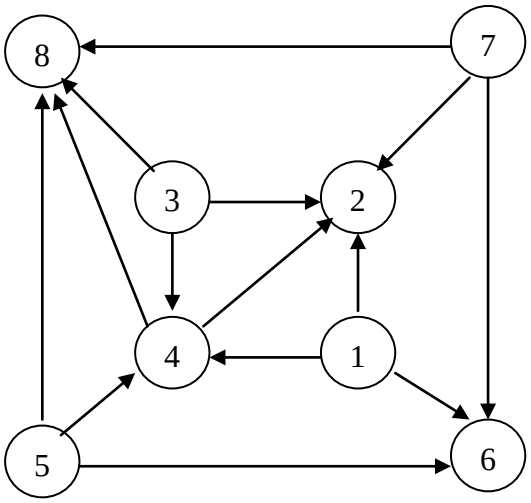
re(it ↑ (f)) =



⇒①, ②, ③, ④ are vertices in a loop.

Exercises:

1. Show that
   - it ↑ (it ↑ (f)) = it ↑ (f)
   - it ↓ (it ↓ (f)) = it ↓ (f)
   - it ↑ (it ↓ (f)) = it ↑ (f)
2. Show that
   if
   it ↑ (f) = f,
   then
   it ↑ (it ↓ (f)) = f.
3. it ↑ (f) is the relation matrix of the minimal transitive relation that contains a given binary relation with the relation matrix f (transitive closure).
   Find in a similar way the maximal transitive relation containing in a given binary relation.
   Example



4. How we can find the rotational direction of a loop?
   How we can separate (color) different disjunct or connected loops in a binary relation?

Part 2: Extensions and applications in the field of mathematics, computer science and algorithm engineering
Summary of part 1:

Input : p ; $\underline{f}$
matrix : = $\underline{f}$
j : = 1(1)p
i : = 1(1)p
yes $f[i, j] \neq 0$ no
$col[i] := col[i] \vee (f[i, j] \wedge) col[j]$
it ↑ ($\underline{f}$) : = matrix
Output : it ↑ ($\underline{f}$) - upwards iteration

Input : p ; $\underline{f}$
matrix : = $\underline{f} \div$ re($\underline{f}$)
j : = 1(1)p
i : = 1(1)p
yes $f[i, j] \neq 0$ no
$col[i] := col[i] \div (f[i, j] \wedge) col[j]$
it ↓ ($\underline{f}$) : = matrix $\vee$ re($\underline{f}$)
Output : it ↓ ($\underline{f}$) - downwards iteration

Two algorithms which throw a long shadow.

Preparation for extension:

Input : p ; f
matrix : = f
j : = 1(1)p
i : = 1(1)p
yes $f[i, j] \neq 0$ no
$col[i] := col[i] \vee (f[i, j] \wedge) col[j]$
it ↑ (f) : = matrix
Output : it ↑ (f) - upwards iteration

The body of the inner for loop

yes $f[i, j] \neq 0$ no
$col[i] := col[i] \vee f[i, j] \wedge col[j]$

is defined over the special algebraic structure

$(\{0,1\}; x \vee y, x \wedge y)$

set — binary algebraic operations, defined on {0, 1}

with:

- $\vee, \wedge : \{0,1\} \times \{0,1\} \rightarrow \{0,1\}$
- $x \vee y$ and $x \wedge y$ satisfy commutative and associative laws
- for $x \vee y$ and $x \wedge y$ exist exactly one identity element, 0 and 1 respectively
- $x \vee y$ is distributive regarding $x \wedge y$
  $x \wedge y$ is distributive regarding $x \vee y$

Now extension:
from the special algebraic structure
$(\{0,1\}; x \vee y, x \wedge y)$
to the generalized algebraic structure
$S = (N; x + y, x \bullet y)$

set "addition" "multiplication"

with

− +, ●: $N \times N \rightarrow N$
− +, ●: commutative and associative
− +, ●: identity elements λ and μ ; $\lambda \bullet x = \lambda$

– +, ●: is distributive over +

Examples:

N	$x + y$	$x \bullet y$	γ	μ	$x - y$
$\{0, 1\}$	$x \vee y$	$x \wedge y$	0	1	$x \div y$
$\{0, 1\}$	$x \oplus y$	$x \wedge y$	0	1	$x \oplus y$
$\mathbb{R}$	$x + y$	$x \cdot y$	0	1	$x - y$
$\mathbb{R}_+$	$\min(x, y)$	$x + y$	∞	0	?

Last step: extension of the inner loop body of the upwards iteration

yes $f[i, j] \neq \lambda$ no	
$col[i] := col[i] + f[i, j] \bullet col[j]$	

Last lecture: extension of the matrix calculus for analysis and synthesis of binary relations on a set

(re(f), sym(f), it↑(f) and it↓(f))

over the Boolean algebra

$(\{0, 1\}; x \vee y, x \wedge y, 0, 1)$

to the algebraic carrier structure

$S = (N; x + y, x \bullet y, \lambda, \mu)$

with

– +, ●: $N \times N \to N$

– +, ●: commutative and associative

– +, ●: identity elements λ and μ; $\lambda \bullet x = \lambda$

– +, ●: is distributive over +

N	$x + y$	$x \bullet y$	γ	μ	$x - y$
$\{0, 1\}$	$x \vee y$	$x \wedge y$	0	1	$x \div y = \begin{cases} 0 & x = y \\ \text{if} & \\ x & x \neq y \end{cases}$
$\{0, 1\}$	$x \oplus y$	$x \wedge y$	0	1	$x \oplus y$
$\mathbb{R}$	$x + y$	$x \cdot y$	0	1	$x - y$
$\mathbb{R}_+$	$\min(x, y)$	$x + y$	∞	0	$\begin{cases} \infty & x = y \\ x \text{ if } & x \neq y \end{cases}$

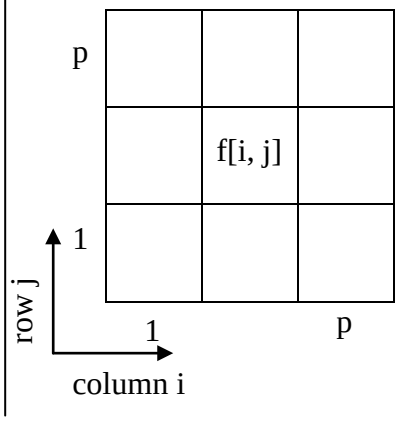
Extension

● Extraction of the main diagonal elements of the $p \times p$ - matrix

$$\underline{f} = \left(f[i, j]\right)_{i=1,\, j=1}^{p\quad p}, \qquad f[i, j] \in N$$

Cartesian notation

(pattern recognition, image processing,...)



over S = (N; +, •, λ, μ)

$$re(\underline{f}) = \begin{pmatrix} \lambda & i \neq j \\ & if \\ f[i, j] & i = j \end{pmatrix}_{i=1,\, j=1}^{p\quad p},$$

● Extraction of the symmetric part of $\underline{f}$ over S

$$sym(\underline{f}) = \begin{pmatrix} \lambda & f[i, j] \neq f[j, i] \\ & if \\ f[i, j] & f[i, j] = f[j, i] \end{pmatrix}_{i=1,\, j=1}^{p\quad p},$$

$\underline{f}$ is ... over S,	if the condition ... is true
reflexive	re($\underline{f}$) = $\underline{e}$ (identity m.)
irreflexive	re($\underline{f}$) = $\underline{0}$ (zero matrix)
symmetric	sym($\underline{f}$) = $\underline{f}$
antisymmetric	sym($\underline{f}$) = re($\underline{f}$)
asymmetric	sym($\underline{f}$) = $\underline{0}$

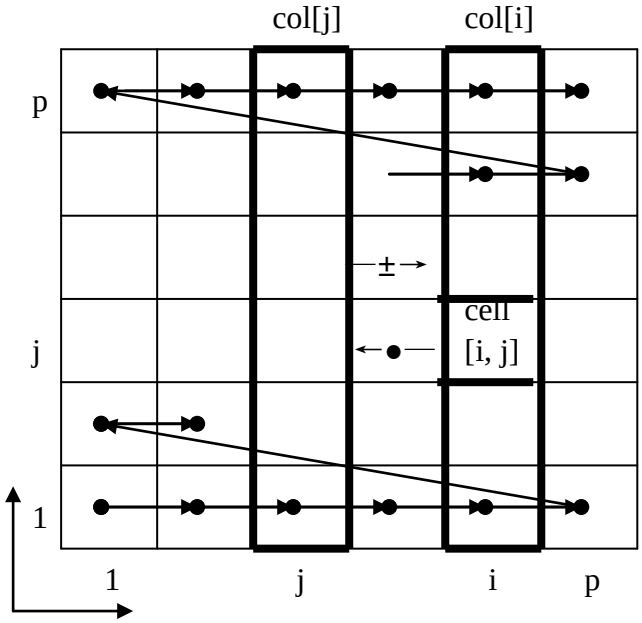
$\underline{e}$ =

λ	λ	..	λ	λ	μ
λ	λ	..	λ	μ	λ
λ	λ		μ	λ	λ
..	..	..		..	..
λ	μ		λ	λ	λ
μ	Λ	..	λ	λ	λ

; $\underline{0}$ =

λ	λ	..	λ	λ	λ
λ	λ	..	λ	λ	λ
λ	λ		λ	λ	λ
..	..	..		..	..
λ	λ		λ	λ	λ
λ	λ	..	λ	λ	λ

● Upwards and downwards iteration of $\underline{f}$ over S



Input : p ; $\underline{f}$
matrix : = $\underline{f}$ / $\underline{f}$ - re($\underline{f}$)
j : = 1(1)p
i : = 1(1)p
yes — $f[i, j] \neq \lambda$ — no
col[i] : = col[i] ± f[i,j] ● col[j]
it ↑ ($\underline{f}$) : = matrix
it ↓ ($\underline{f}$) : = matrix + re($\underline{f}$)
Output : it ↑ ($\underline{f}$) – upwards iteration it ↓ ($\underline{f}$) – downwards iteration

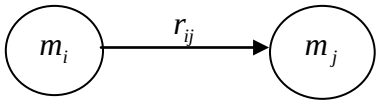
Now we test the extension for

- $S = (N; +, \bullet, \lambda, \mu) = (\mathbb{R}[0,\infty]; \min, +, \infty, 0)$
- with the shortest paths problem.

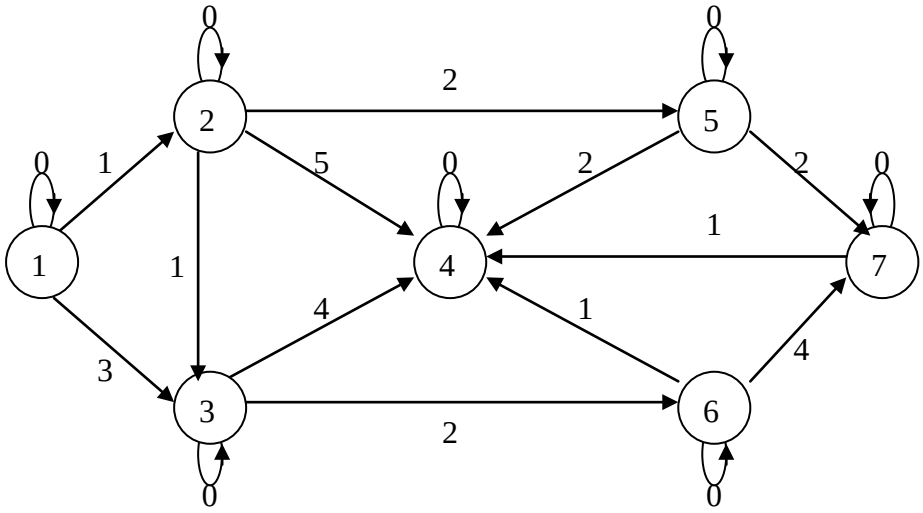
Given a directed graph with

- $M = \{m_1, m_2, ..., m_p\}$ – set of the vertices
- $\underline{f} = \left(f[m_i, m_j]\right)_{i=1,\, j=1}^{p\quad p}$ – matrix of the direct distances $f(m_i, m_j) \in \Re[0,\infty]$ from each vertex m_i to each other vertex m_j .

  $f(m_i, m_j) = \infty$ – there is no direct way (edge) from vertex m_i to m_j , m_j is not directly reachable from m_i .

$f(m_i, m_j) = r_{ij} \in \mathbb{R}[0,\infty]$	
$f(m_i, m_j) = \lambda = \infty$	

Example



- $M = \{1, 2, 3, 4, 5, 6, 7\}$

$-\underline{f} =$

7	∞	∞	∞	∞	2	4	0
6	∞	∞	2	∞	∞	0	∞
5	∞	2	∞	∞	0	∞	∞
4	∞	5	4	0	2	1	1
3	3	1	0	∞	∞	∞	∞
2	1	0	∞	∞	∞	∞	∞
1	0	∞	∞	∞	∞	∞	∞
j / i	1	2	3	4	5	6	7

Find the matrix

$$\underline{\varphi} = \left(\varphi(m_i, m_j)\right)_{i=1,\, j=1}^{p \quad p}$$

of the shortest distances (paths) $\phi(m_i, m_j) \in \mathbb{R}[0, \infty]$ from each vertex m_i to each other vertex m_j .

$\varphi(m_i, m_j) = \infty$ - there is no direct or indirect way (edge or sequence of edges) from vertex m_i to vertex m_j , m_j is neither directly nor indirectly reachable from m_i .

Solution

$$\underline{\varphi} = it \uparrow (\underline{f})$$

For our example is

$\underline{\varphi} = it \uparrow (\underline{f}) =$

7	5	4	6	∞	2	4	0
6	4	3	2	∞	∞	0	∞
5	3	2	∞	∞	0	∞	∞
4	5	4	3	0	2	1	1
3	2	1	0	∞	∞	∞	∞
2	1	0	∞	∞	∞	∞	∞
1	0	∞	∞	∞	∞	∞	∞
	1	2	3	4	5	6	7

Comment 1

Because of min(x, x) = x is $it \uparrow (\underline{\varphi}) = \underline{\varphi}$

Comment 2

A nonnegative real function f(x, y) on a set M = {m_1, m_2,..., m_p}

$$f : M \times M \to \mathbb{R}[0, \infty]$$

with

re($\underline{f}$) = $\underline{e}$

sym($\underline{f}$) = $\underline{f}$

it↑(f) = f
over $S = (\mathbb{R}[0,\infty]; \min, +, \infty, 0)$
is called metric on M.
All direct distances in the graph of f are here at the same time also the shortest distances.
Example
M = {m_1, m_2,..., m_8}

f =

8	23	20	19	18	14	11	12	0
7	11	8	7	8	2	1	0	12
6	12	9	8	7	3	0	1	11
5	9	6	5	8	0	3	2	14
4	5	2	3	0	8	7	8	18
3	4	1	0	3	5	8	7	19
2	3	0	1	2	6	9	8	20
1	0	3	4	5	9	12	11	23
	1	2	3	4	5	6	7	8

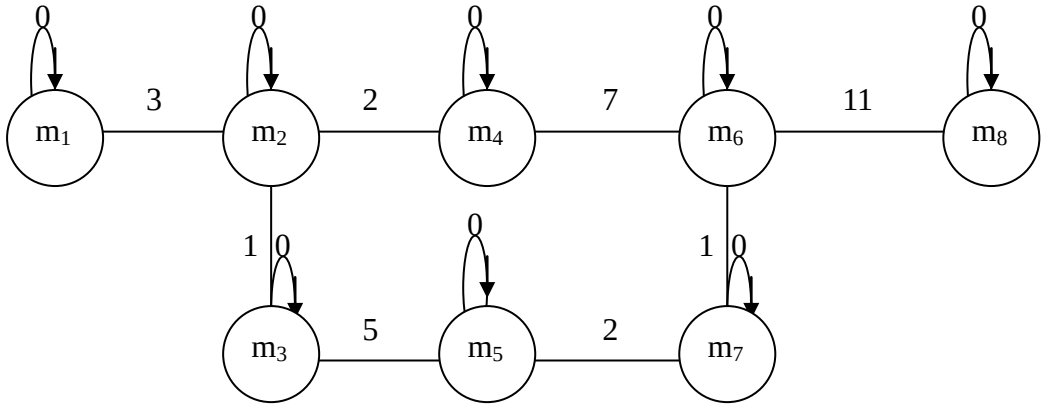
re(f) = e f is reflexive
sym(f) = f symmetric
it↑(f) = f transitive
over $S = (\mathbb{R}[0,\infty]; \min, +, \infty, 0)$
f is metric on M.
How about it↓(f)?

φ = it↓(f) =

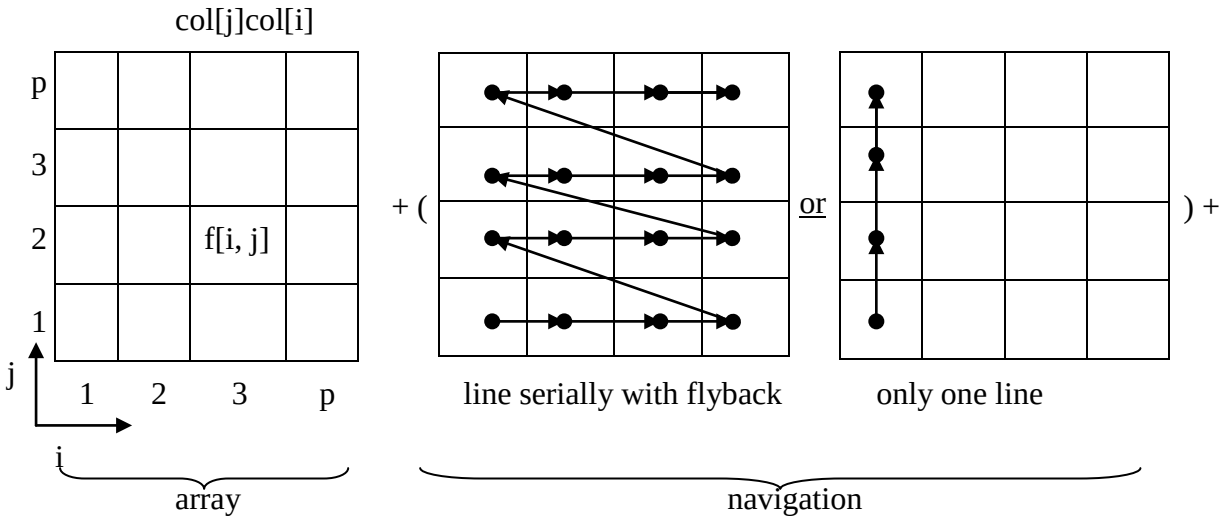
8	∞	∞	∞	∞	∞	11	∞	0
7	∞	∞	∞	∞	2	1	0	∞
6	∞	∞	∞	7	∞	0	1	11
5	∞	∞	5	∞	0	∞	2	∞
4	∞	2	∞	0	∞	7	∞	∞
3	∞	1	0	∞	5	∞	∞	∞
2	3	0	1	2	∞	∞	∞	∞
1	0	3	∞	∞	∞	∞	∞	∞
	1	2	3	4	5	6	7	8

re(φ) = e φ is reflexive

sym(φ) = f symmetric
it↓(φ) = φ intransitive
over $S = (\mathbb{R}[0,\infty]; \min, +, \infty, 0)$
φ – metric skeleton; it↑(φ) = f
measuring scale



Summary
Matrix algorithms for analysis and synthesis of binary relations on a set are the basis for extension and many new applications.
These matrix algorithms consist of 3 parts:
“array + navigation + local operator”, in our case:



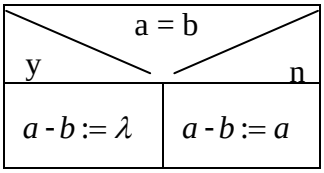
$$"(\forall i)(\forall j) col[i] := col[i] \pm f[i, j] \bullet col[j]"$$

local operator

The navigation is controlled here from outside by a given program (f.e. “line serially with flyback” or “only one line”), another possibility is the data controlled navigation. In this case the local operator is mostly local calculator and at the same time also local navigator (go to next position). The local operator is here a triple operator (col[i], f[i, j], col[j]). For many applications, especially in the field of picture processing, local operators with a larger data intake area are in use.

Input: matrix $\underline{f} = (f[i, j])_{i=1,\, j=1}^{p\quad p}$

Output: matrix $\underline{\varphi} = \begin{cases} it\uparrow(\underline{f}) = \uparrow \underline{f} & + \\ & for \\ it\downarrow(\underline{f}) = \downarrow \underline{f} & - \end{cases}$

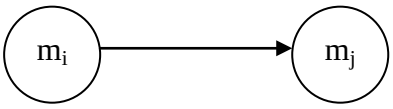
nr.	operand set	algebraic operations				
		a + b	a • b	λ	μ	a – b
1	{0, 1}	$a \vee b$	$a \wedge b$	0	1	$a \dot{-} b$ 
2	$\mathbb{R}[0,\infty]$	$\min(a,b)$ $a \perp b$	$a+b$	∞	0	
3	$\mathbb{R}$	$a+b$	$a \cdot b$	0	1	$a-b$
4	{0, 1}	$a \oplus b$	$a \wedge b$	0	1	$a \oplus b$

Applications

Nr. 1 $a + b = a \vee b$; $a \bullet b = a \wedge b$

1.1 f is adjacency matrix of a graph on M

f[i, j] = 1 m_i is (directly) related to m_j



from vertex m_i leads an edge (a direct way) to vertex m_j ;
vertex m_i has direct access to vertex m_j ;
vertex m_j is directly accessible (reachable) from vertex m_i

f[i, j] = 0 no

⇩

φ = ↑ f – accessibility (reachability) matrix of this graph on M

φ[i, j] = 1 m_i is directly or indirectly related to m_j

vertex m_i has direct or indirect access to vertex m_j

vertex m_j is directly or indirectly accessible (reachable) from vertex m_i

1.2 Loop finder in a graph on M

f is adjacency matrix of a graph.

with re(f) = 0; sym(f) = 0.

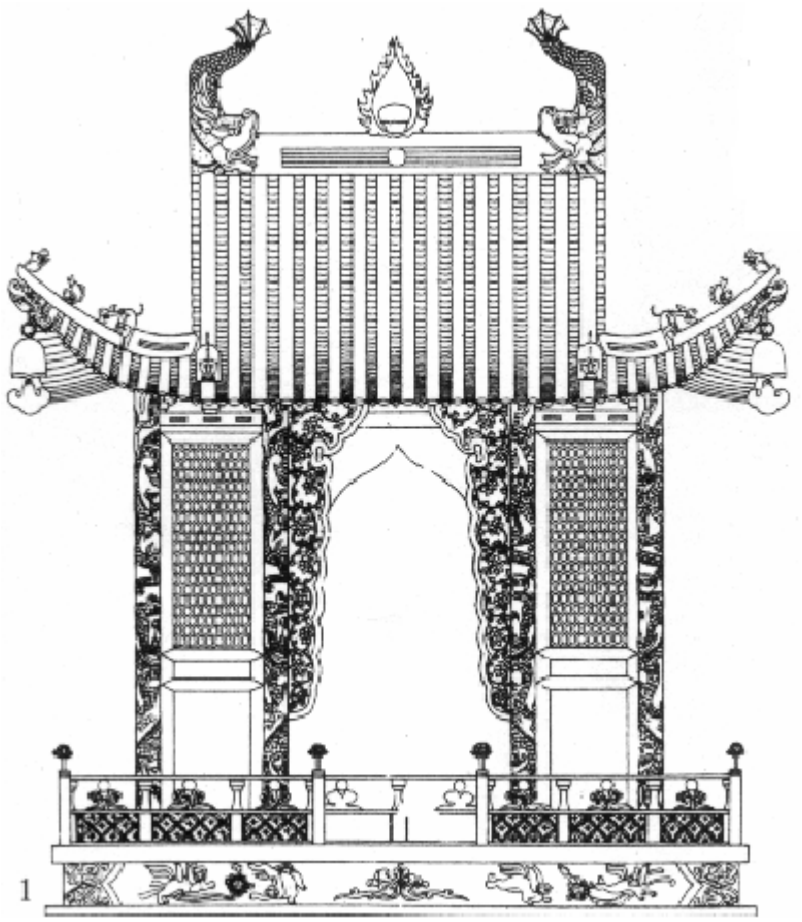
Then indicates

re(↑ f) = 0 no loops

re(↑ f) ≠ 0 one or more loops.

How we can find the rotational direction of a loop?

How we can separate (color) different disjunct or connected loops in this graph?



1

Nr.2 $a + b = \min(a, b) = a \perp b$; $a \cdot b = a + b$

$\underline{f}$ is matrix of direct distances of the vertices of a graph

$f[i, j] = r_{ij}$ – there is a direct path from vertex m_i to vertex m_j with the length r_{ij}.

$f[i, j] = \infty$ – there is not a direct path from m_i to m_j

⇓

$\underline{\varphi} = \uparrow \underline{f}$ – matrix of the shortest-path distances of the vertices of this graph.



Nr.3

3.1 $\underline{f}$ – product structure matrix of a machine, device,...

f[i, j] = n_{ij} component m_i is directly built in component m_j n_{ij} times

f[i, j] = 0 component m_i is not directly built in component m_j

⇩

$\underline{\varphi} = \uparrow \underline{f}$ – requirement matrix of this product

φ[i, j] = N_{ij} component m_i is cumulatively (altogether) built in component m_j N_{ij} times

$\underline{f}$ =

12												
	0											0
							1					1
								8	1			
			3	3			1					
					1							
			5		5	1						
j 1	0	2	4	4							1	0
	1											12

i

; $\underline{\varphi} = \uparrow \underline{f}$ =

0											0
					1	8	1				1
							8	1			
		3	3	3		1	8	1			
				1							
		8	3	8	1	1	8	1			
0	2	20	10	20	2	2	16	2		1	0

3.2 Continued fraction

$$a_0 + \cfrac{b_1}{a_1 + \cfrac{b_2}{a_2 + \cfrac{b_3}{a_3 + \ddots + \cfrac{b_n}{a_n}}}} = \frac{B_n}{A_n}$$

f =

	1	2	3	4	5	6	7
7					b_n	a_n	
6							
5			b_3	a_3			
4		b_2	a_2				
3	b_1	a_1					
2	a_0						
1	0						

j

i

	1	2	3	4	5	6	7
7						6	6
6					5	5	
5				4	4		
4		3	3				
3	2	2					
2	2						
1	0						

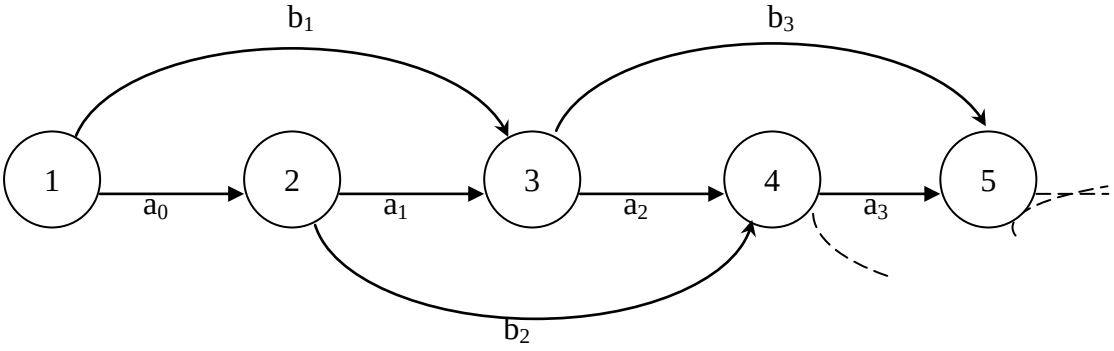
φ = ↑ f = only col[1] and col[2]

	1	2	3	4	5	6	7
7	B_n	A_n					
6							
5							
4	B_2	A_n					
3	B_1	A_1					
2	a_0						
1	0						

j

i

	1	2
7	5040	1854
6	720	265
5	120	44
4	24	9
3	6	2
2	2	
1	0	





3.3 Horner's problem, Horner's algorithm (rule)

Given polynomial in x of order n with constant coefficients

$$p_n(x) = \sum_{i=0}^{n} a_i x^{n-i} \qquad (1)$$

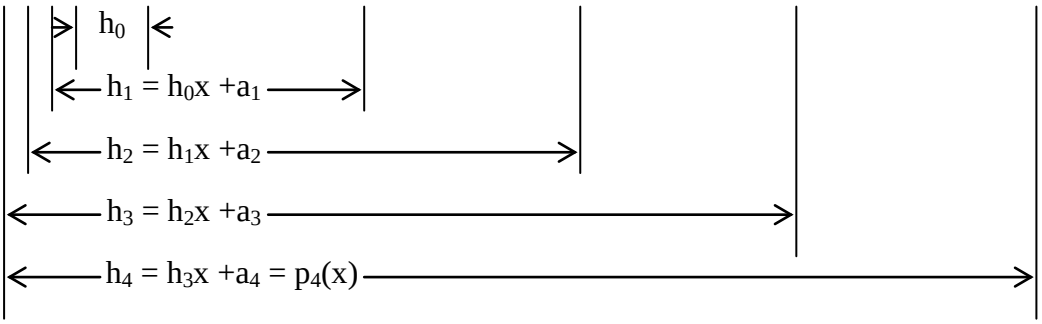
$$n = 4 \rightarrow p_4(x) = a_0 x^4 + a_1 x^3 + a_2 x^2 + a_3 x + a_4 \qquad (2)$$

Find the value of $p_n(x)$ for given a_i and a given value of the variable x in the computationally most efficient way (minimal number of multiplications).

Solution: Horner's algorithm (rule)

- Transform (1)/(2) into a nested expression:

$$(((a_0) \cdot x + a_1) \cdot x + a_2) \cdot x + a_3) \cdot x + a_4$$

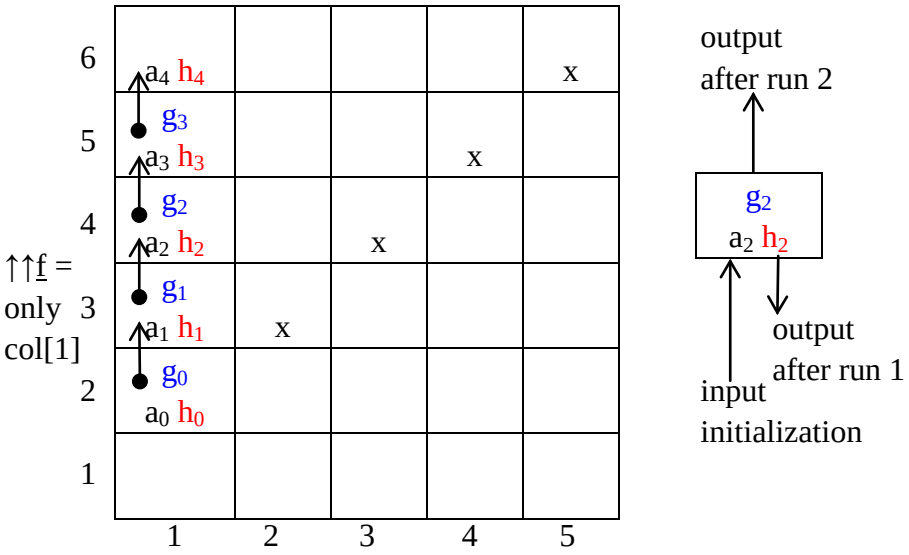


- Algorithm executed by upwards iteration

  $h_0 = a_0$

  $h_{i+1} = h_i \cdot x + a_{i+1}$; i = 0(1)n-1

  $p_n(x) = h_n$



- Extension

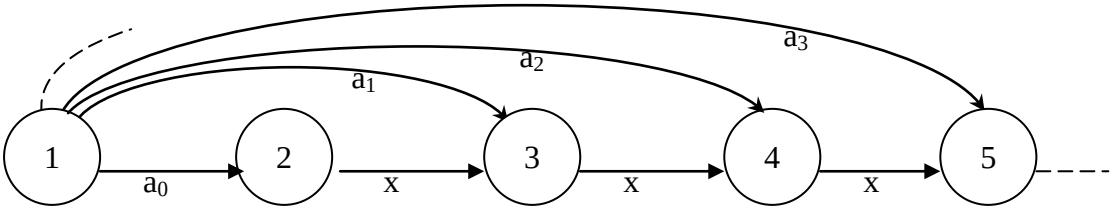
  Repeat the upwards iteration:

  $g_0 = h_0$

  $g_{i+1} = g_i \bullet x + h_{i+1}$; i = 0(1)?

  $g_3 = p'_4(x)$

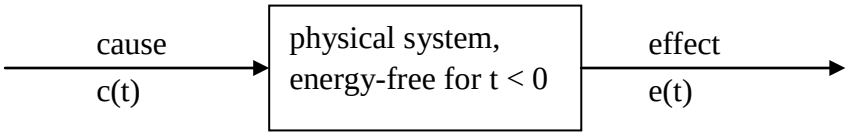
  The result of the first run is the value of $p_n(x)$; the result of the second run is the value of $p'_n(x)$.



How we can calculate in the same way

$\left(p'_n(x)\right)^2 - p_n(x) \cdot \frac{1}{2} p''_n(x)$ etc?

3.4 **Given** a

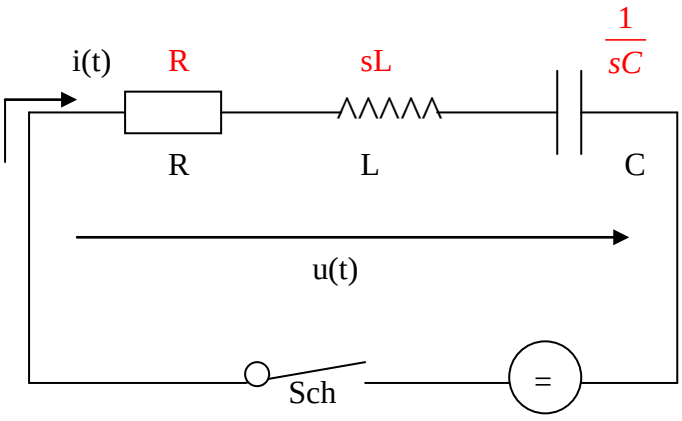


whose dynamic behavior is described by a linear ordinary differential equation with constant coefficients

$$\sum_{i=0}^{n} a_i e^{(i)}(t) = \sum b_j c^{(j)}(t); \quad a_n = 1, \quad n \geq m$$

$$e^{(i)} = \frac{d^i c(t)}{dt^i}; \quad c^{(j)} = \frac{d^j c(t)}{dt^j}$$

Example: electrical oscillatory circuit



energy-free for t < 0

→ $i_L(0) = i(0) = 0$

$u_C(0) = 0$

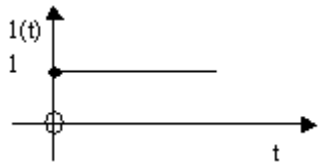
$$U = (R + sL + \frac{1}{sC})J \rightarrow (s^2 + \frac{R}{L}s + \frac{1}{LC})J = \frac{1}{L}sU$$

$$\frac{d^2 i(t)}{dt^2} + \frac{R}{L} \cdot \frac{di(t)}{dt} + \frac{1}{LC} \cdot i(t) = \frac{1}{L}\frac{du(t)}{dt}$$

$$n = 2;\ \ a_2 = 1,\ \ a_1 = \frac{R}{L},\ \ a_0 = \frac{1}{LC}$$

$$m = 1;\ \ b_2 = 0,\ \ b_1 = \frac{1}{L},\ \ b_0 = 0$$

Find a computationally efficient power series in t representation of the step response esr(t) = e(t) for c(t) = c(0)•1(t)



of this physical system.

Solution

$$esr(t) = c(0) \cdot \sum_{i=0}^{\infty} \frac{d_i}{i!} t^i;\ \ t \geq 0$$

6	d_4 b_{n-4}	a_{n-4}	a_{n-3}	a_{n-2}	a_{n-1}	0
5	d_3 b_{n-3}	a_{n-3}	a_{n-2}	a_{n-1}	0	0
4	d_2 b_{n-2}	a_{n-2}	a_{n-1}	0	0	0
3	d_1 b_{n-1}	a_{n-1}	0	0	0	0
2	d_0 output b_n input	0	0	0	0	0
1	0	0	0	0	0	0
	1	2	3	4	5	6

$d_2 = b_{n-2} - b_n a_{n-2} - (b_{n-1} - b_n a_{n-1})a_{n-1}$

$d_1 = b_{n-1} - b_n a_{n-1}$

$d_0 = b_n$

↓f = only col[1]

What delivers ↓↓ *f* ?

For our example is

6	d_4 0			$\frac{1}{LC}$	$\frac{R}{L}$	0	$d_4 = L\frac{R}{L^3C} - \frac{R^3}{L^4}$
5	d_3 0		$\frac{1}{LC}$	$\frac{R}{L}$	0		$d_3 = \frac{R^2}{L^2} - \frac{1}{L^2C}$
4	d_2 0	$\frac{1}{LC}$	$\frac{R}{L}$	0			$d_2 = -\frac{R}{L^2}$
↓ f = only col[1] 3	d_1 $\frac{1}{L}$	$\frac{R}{L}$	0				$d_1 = \frac{1}{L}$
2	d_0 0 output input 0						$d_0 = 0$
1	0						
	1	2	3	4	5	6	

$$i(t) = \frac{u(0)}{R}\left(\underbrace{\frac{1}{1!}\left(\frac{t}{\tau}\right)^1 - \frac{1}{2!}\left(\frac{t}{\tau}\right)^2}_{\text{start of i(t) depends only from R and L!}} + \frac{1}{3!}(1-\delta)\left(\frac{t}{\tau}\right)^3 - \frac{1}{4!}(1-2\delta)\left(\frac{t}{\tau}\right)^4 + \frac{1}{5!}(1-3\delta+\delta^2)\left(\frac{t}{\tau}\right)^5 \mp \ldots\right)$$

$$\tau = \frac{L}{R};\quad \delta = \tau^2\frac{1}{LC}$$

An “algorithmic microscope”: what happens immediately after t = + 0?

The Euler transformation

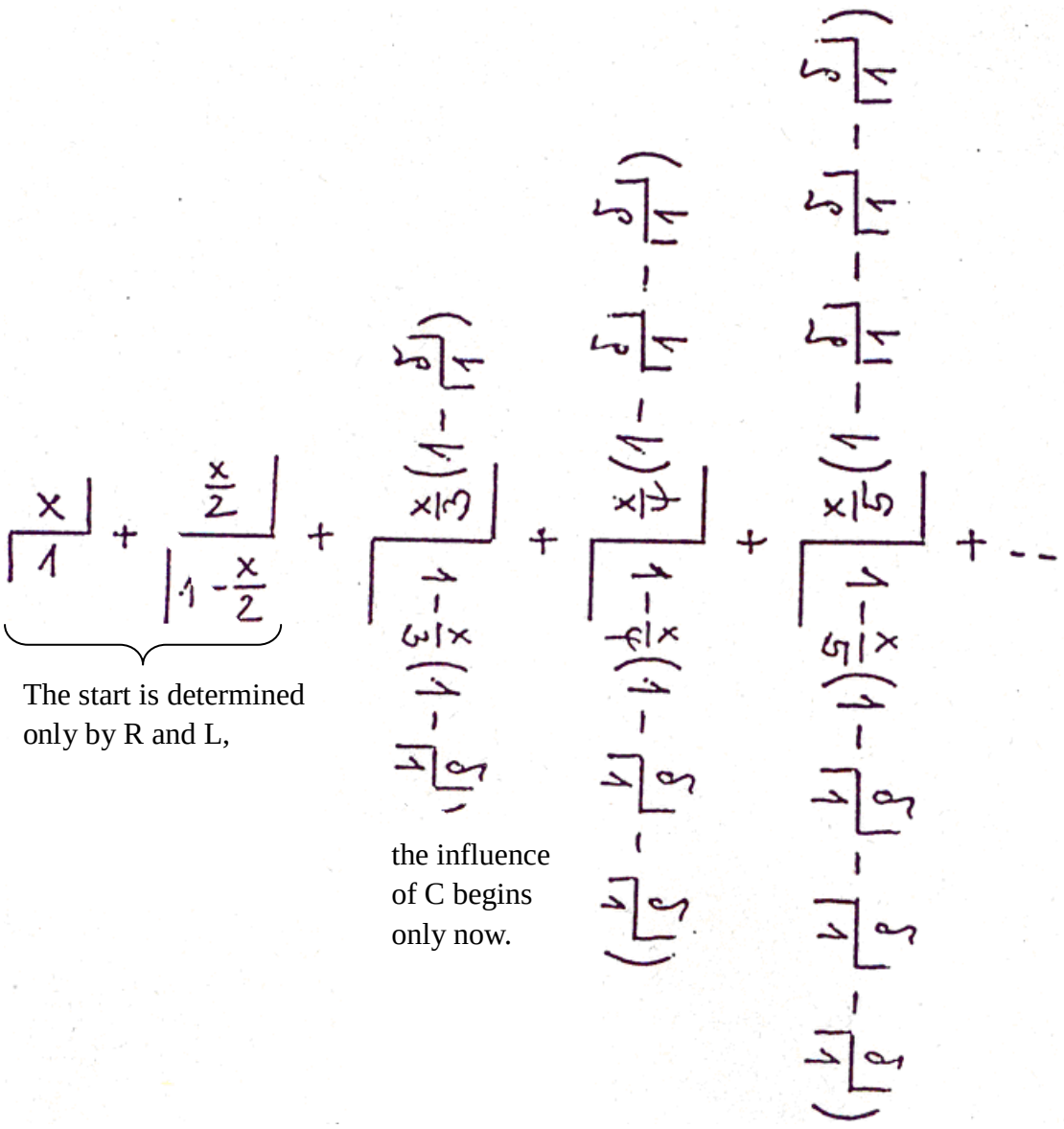
$$\sum_{i=0}^{\infty} a_i \cdot x^i = a_0 + \left|\begin{array}{l} \frac{a_1 x}{1} + \left|\begin{array}{l} \frac{-\frac{a_i}{a_{i-1}}x}{1+\frac{a_i}{a_{i-1}}x} \end{array}\right| \end{array}\right| \quad \text{i = 2, 3, 4, ...}$$

of the i(t) power series with

$$\tau = \frac{L}{R};\quad \delta = \tau^2\frac{1}{LC};\quad x = \frac{t}{\tau}$$

into a continued fraction results in the “score of the i(t) concert”

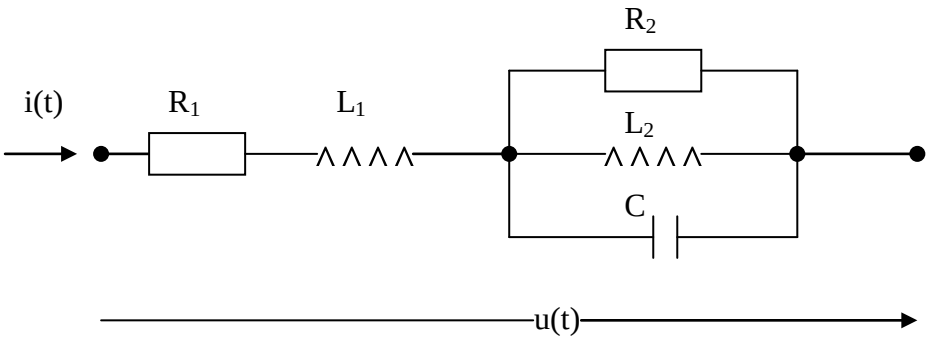
$$\frac{i(t)_{|u(t)=u(0)1(t)}}{\frac{u(0)}{R}} =$$



The start is determined only by R and L,

the influence of C begins only now.

Find in the same way the “continued fraction score” for the step response $i(t)_{|u(t)=u(0)1(t)}$ of the electrical equivalent circuit of a loudspeaker.



3.5 **Given** two real arbitrarily often differentiable functions u(x) and v(x).
Find an efficient algorithm for calculation of the coefficients

$$a_i = \frac{1}{i!} \cdot \frac{d^i}{dx^i}\left(\frac{u(x)}{v(x)}\right); \quad x = a$$

of the Taylor series representation of $\frac{u(x)}{v(x)}$ about x = a : $\frac{u(x)}{v(x)} = \sum_{i=0}^{\infty} a_i \cdot (x-a)^i$.

Solution

$$a_i = \frac{b_i}{(v(a))^{i+1}}; \quad U^{(i)} = \frac{1}{i!} \cdot \frac{d^i u(x)}{dx^i}; \; x = a$$

$$V^{(i)} = \frac{1}{i!} \cdot \frac{d^i v(x)}{dx^i}; \; x = a$$

6	b_4 $U^{(4)}$	$U^{(4)}$	U'''	U''	U'	V
5	b_3 U'''	U'''	U''	U'	V	0
4	b_2 U''	U''	U'	V	0	0
3	b_1 U'	U'	V	0	0	0
2	b_0 U output / input V		0	0	0	0
1	0	0	0	0	0	0
	1	2	3	4	5	6

↓ f = only col[1]

$b_2 = (U''V - UV'') - (U'V - UV')V'$

$b_1 = U'V - UV'$

$b_0 = U$

Extension of the tripel operator $col[i] := col[i] \pm f[i, j] \bullet col[j]$
to the quadrupel operator $col[i] := f[j, j] \bullet col[i] \pm f[i, j] \bullet col[j]$; $k > j$

Example 1

$$u(x) = 1 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \cdots$$

$$v(x) = 1 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \cdots$$

$$a = 0$$

$$\frac{u(x)}{v(x)} = \sum_{i=0}^{\infty} b_i x^i$$

6	b_4 α_4	β_4	β_3	β_2	β_1	1
5	b_3 α_3	β_3	β_2	β_1	1	0
4	b_2 α_2	β_2	β_1	1	0	0
3	b_1 α_1	β_1	1	0	0	0
2	b_0 output 1 input	1	0	0	0	0
1	0	0	0	0	0	0

↓ f = only col[1]

$$b_0 = 1;\ b_1 = \alpha_1 - \beta_1;\ b_2 = \alpha_2 + \beta_1^2 - \alpha_1\beta_1 - \beta_2$$

$$b_3 = \alpha_3 + \alpha_1\beta_1^2 + 2\beta_1\beta_2 - \alpha_1\beta_2 - \alpha_2\beta_1 - \beta_1^3 - \beta_3;\ \ldots$$

Example 2

$$u(x) = \sin x;\quad v(x) = \cos x;\quad a = 0$$

$$\frac{u(x)}{v(x)} = \frac{\sin x}{\cos x} = \tan x = \sum_{i=1}^{\infty} (-1)^{i+1} b_i x^{2i-1}$$

1 2 3 4 5 6

6	b_5 $\frac{1}{5!}$	$\frac{1}{8!}$	$\frac{1}{6!}$	$\frac{1}{4!}$	$\frac{1}{2!}$	$\frac{1}{0!}$
5	b_4 $\frac{1}{7!}$	$\frac{1}{6!}$	$\frac{1}{4!}$	$\frac{1}{2!}$	$\frac{1}{0!}$	0
4	b_3 $\frac{1}{5!}$	$\frac{1}{4!}$	$\frac{1}{2!}$	$\frac{1}{0!}$	0	0
3	b_2 output $\frac{1}{3!}$ input	$\frac{1}{2!}$	$\frac{1}{0!}$	0	0	0
2	b_1 $\frac{1}{1!}$	$\frac{1}{0!}$	0	0	0	0
1	0	0	0	0	0	0

↓ f = only col[1]

$$b_3 = \frac{1}{0!}\cdot\frac{1}{0!}\cdot\frac{1}{5!} + \frac{1}{1!}\cdot\frac{1}{2!}\cdot\frac{1}{2!} - \frac{1}{0!}\cdot\frac{1}{1!}\cdot\frac{1}{4!} - \frac{1}{0!}\cdot\frac{1}{2!}\cdot\frac{1}{3!}$$

$$b_2 = \frac{1}{0!}\cdot\frac{1}{3!} - \frac{1}{1!}\cdot\frac{1}{2!}$$

$$b_1 = \frac{1}{1!}$$

$$\tan x = \sum_{i=1}^{\infty} (-1)^{i+1} b_i x^{2i-1}$$

$$\tanh x = \sum_{i=1}^{\infty} b_i x^{2i-1}$$

$$\frac{x}{\tan x} - 1 = \sum_{i=1}^{\infty} (-1)^{i+1} \frac{b_i}{2^{2i} - 1} x^{2i}$$

$$\frac{x}{\tanh x} - 1 = -\sum_{i=1}^{\infty} \frac{b_i}{2^{2i} - 1} x^{2i}$$

Example 3

$$u(x) = 1; \quad v(x) = \cos x; \quad a = 0$$

$$\frac{u(x)}{v(x)} = \frac{1}{\cos x} = 1 + \sum_{i=1}^{\infty} (-1)^{i+1} b_i x^{2i}$$

$$\frac{1}{\cosh x} = 1 - \sum_{i=1}^{\infty} b_i x^{2i}$$

6	b_5 $\frac{1}{10!}$	$\frac{1}{8!}$	$\frac{1}{6!}$	$\frac{1}{4!}$	$\frac{1}{2!}$	$\frac{1}{0!}$
5	b_4 $\frac{1}{8!}$	$\frac{1}{6!}$	$\frac{1}{4!}$	$\frac{1}{2!}$	$\frac{1}{0!}$	0
4	b_3 $\frac{1}{6!}$	$\frac{1}{4!}$	$\frac{1}{2!}$	$\frac{1}{0!}$	0	0
3	b_2 output $\frac{1}{4!}$ input	$\frac{1}{2!}$	$\frac{1}{0!}$	0	0	0
2	b_1 $\frac{1}{2!}$	$\frac{1}{0!}$	0	0	0	0
1	0	0	0	0	0	0
	1	2	3	4	5	6

↓ f= only col[1]

$$b_3 = \frac{1}{0!}\cdot\frac{1}{0!}\cdot\frac{1}{6!} + \frac{1}{2!}\cdot\frac{1}{2!}\cdot\frac{1}{2!} - \frac{1}{0!}\cdot\frac{1}{2!}\cdot\frac{1}{4!} - \frac{1}{0!}\cdot\frac{1}{2!}\cdot\frac{1}{4!}$$

$$b_2 = \frac{1}{0!}\cdot\frac{1}{4!} - \frac{1}{2!}\cdot\frac{1}{2!}$$

$$b_1 = \frac{1}{2!}$$

Find time to think

In general, polynomial equations

$$p_n(x) = \sum_{i=0}^{n} a_i x^i = a_n \prod_{j=1}^{n} (x - x_j) = 0;\ n \geq 5 \qquad (1)$$

can not be solved (given the coefficients a_i, find the zero´s x_j) in expressions containing the coefficients a_i and a finite number of additions, subtractions, multiplications, divisions, and root extractions.
But in the special case (wormhole)

$$p_n(x) = \sum_{i=0}^{n} a_i x^i = a_n \prod_{j=1}^{k} (x - x_j)^{v_j} = 0;\ \ \sum_{j=1}^{k} v_j = n;$$

all x_j and v_j are pairwise different from each other, respectively (2)

even a solution in rational expressions containing the a_i and a finite number of additions, subtraction, multiplications and divisions is possible.
Find a simple algorithms to solve the special case equ.(2) in this way.

Solving advises to equ.(2):

- $$-\frac{a_{n-1}}{a_n} = \sum_{j=1}^{k} v_j x_j \qquad (3)$$

- $$\gcd(p_n(x), p_n'(x)) = \prod_{j=1}^{k} (x - x_j)^{v_j - 1} \qquad (4)$$

$\gcd(p_n(x), p_n'(x))$ – greatest common divisor of the polynomials $p_n(x)$ and $p_n'(x)$.
Find $\gcd(p_n(x), p_n'(x))$ using the algorithmic system “array (initialized with the a_i) + navigation + local operator”.

Transitive relation

http://en.wikipedia.org/wiki/Transitive_relation