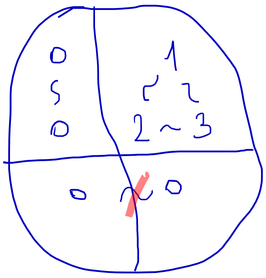
Ekvivalence

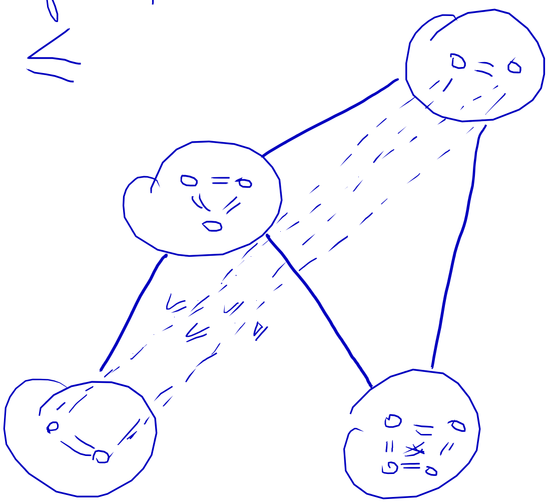
Rozklad množiny M podle relace ~ / faktorová množina podle ~



$[1]_\equiv = \{1,2,3\}$

Kvaziuspořádání (preorder)
reflex., tranzit.

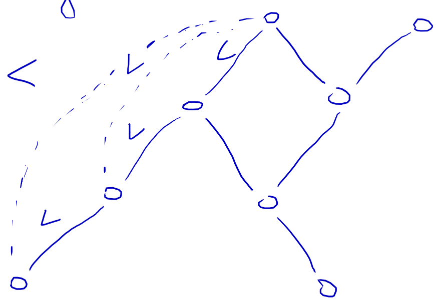
$\leq$



Částečné uspořádání (partial order poset)

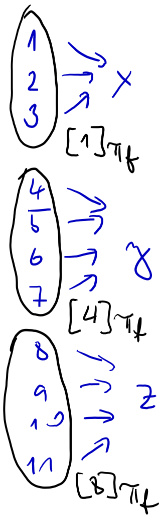
Hasseův diagram

$<$



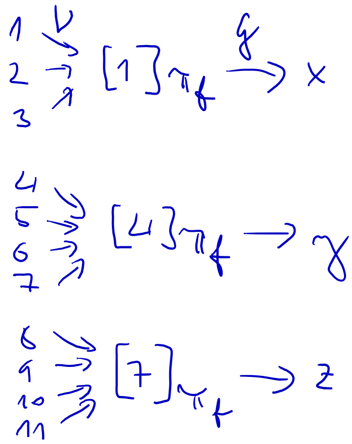
---

jádro zobrazení f − π_f



kanonické zobrazení

$f = g \circ \nu$



---

kongruence $\equiv$, $\mathcal{A} = (A, *)$

$\forall a, b, a', b' \in A : a \equiv a' \wedge b \equiv b' \Rightarrow a * b \equiv a' * b'$

homomorfismus $h : \mathcal{A} \to \mathcal{B}$, $\mathcal{B} = (B, +)$

$\forall a, b \in A \quad h(a * b) = h(a) + h(b)$

$(\mathbb{Z}, +, *)$ $\quad n \in \mathbb{Z}$ $\quad (\langle 0, n-1\rangle, +_n, *_n)$ zbytková třída s modulem n

$$a +_n b = (a+b) \% n$$
$$a *_n b = (a*b) \% n$$

$$h : \mathbb{Z} \to \langle 0, n-1\rangle$$
$$a \mapsto a \% n$$

Je h homom.?

1) $h(a+b) \stackrel{?}{=} h(a) +_n h(b)$ ✓

2) $h(a*b) \stackrel{?}{=} h(a) *_n h(b)$ ✓

víme, že $\forall a, b$

$a = xn + a \% n$

$b = yn + b \% n$

$$h(a+b) = (a+b) \% n = (xn + a\%n + yn + b\%n) \% n =$$
$$= (a\%n + b\%n) \% n = h(a) +_n h(b)$$

$$h(a*b) = (a*b) \% n = ((xn + a\%n) * (yn + b\%n)) \% n =$$
$$= (xnyn + xn(b\%n) + (a\%n)yn + (a\%n)(b\%n)) \% n =$$
$$= (a\%n * b\%n) \% n = h(a) *_n h(b)$$

$\equiv_n$ - jádro homom. h (čísla se stejným zbytkem modulo n)

$a \mapsto [a]_{\equiv_n}$ ... přiroz. homomorfismus

$\{[0]_{\equiv_n}, [1]_{\equiv_n}, \ldots, [n-1]_{\equiv_n}\}$ rozklad $\mathbb{Z}$ podle $\equiv_n$

$\equiv_n$ je kongruence $\quad a \equiv_n a' \wedge b \equiv_n b' \implies a+b \equiv_n a' +_n b' \wedge a*b \equiv_n a' *_n b'$

faktor algebra podle $\equiv_n$

$$[a]_{\equiv_n} + [b]_{\equiv_n} = [a+b]_{\equiv_n}$$
$$[a]_{\equiv_n} * [b]_{\equiv_n} = [a*b]_{\equiv_n}$$

(je možné počítat na libovolných reprezentantech tříd)

Co zaokrouhlování, ořezávání desetinných míst?

$$\underset{0{,}4\ \ 0{,}4}{round(a+b)} \overset{?}{=} \underset{0{,}4}{round(a)} + \underset{0{,}4}{round(b)}$$

$$1 \neq 0 \quad 0$$

round není homom.

podobně
$round(1{,}4 * 1{,}4) \neq round(1{,}4) * round(1{,}4)$

} důsledek: Chyba se akumuluje

$$\underset{2{,}25}{floor(1{,}5 * 1{,}5)} = floor(1{,}5) * floor(1{,}5)$$

$$\neq \quad 1 \quad 1$$

floor není homom

Nechť $(R, \cdot, +, {}^*)$ je algebra regulárních výrazů
$(A, \cdot_A, +_A, {}^{*_A})$ algebra automatů s automatovými operacemi implementujícími $\cdot, +, {}^*$ (viz. Opora TIN)
$(\mathcal{L}, \cdot_L, +_L, {}^{*_L})$ algebra jazyků s jazykovými operacemi

Nechť $h: R \to A$ je převod regulárního výrazu na automat (z Opory).
h je homomorfismus
$$h(r \cdot r') = h(r) \cdot_A h(r')$$
$$h(r + r') = h(r) +_A h(r')$$
$$h(r^*) = h(r)^{*_A}$$

Relace $\equiv \subseteq R \times R$ s.t.
$r \equiv r' \Leftrightarrow L(r) = L(r')$ je kongruence: $L(r)$ je jazyk repres. výrazem r

$r \equiv r' \wedge s \equiv s' \Rightarrow L(r \cdot s) = L(r' \cdot s') \Rightarrow r \cdot s \equiv r' \cdot s'$

a podobně pro $+$ a *

Je $\equiv$ jádrem h? Tj., platí, že reg. v. jsou převedeny na stejný automat právě když reprezentují stejný jazyk?

1) Nechť $h(r) = h(r')$. Potom zřejmě $L(r) = L(r')$, tj. $r \equiv r'$. ✓

2) Nechť $r \equiv r'$. Platí, že $h(r) = h(r')$? ✗

Ne! Protipříklad: $r = \varepsilon + a \cdot a^*$, $r' = a^*$. Takže $\equiv$ není jádrem h.

Homomorphic encryption

Chci počítat, ale nechci, aby procesor znal má data. násobení v matic

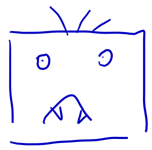
Chci Π data

$\downarrow h$

encryption $h(\text{data})$

$\downarrow$

Σ



$\{\Sigma$

$\downarrow$

$\Sigma(h(\text{data}))$

$\downarrow h^{-1}$

decryption $h^{-1}(\Sigma(h(\text{data}))) = \Pi$ data

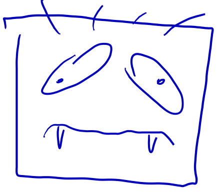
Chci Σ data

$\downarrow g$

$g(\text{data})$

$\downarrow$

Π



$\{\Sigma$

$\downarrow$

$\Pi(g(\text{data}))$

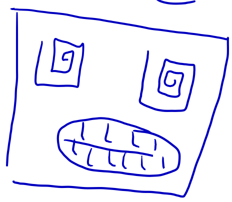
$\downarrow g^{-1}$

$g^{-1}(\Pi(g(\text{data}))) = \Sigma$ data

Chci Π data

$\downarrow f$

Π matic



podobně

encr. je násobení permut. maticí,

decr. je násobení její inverzí

$h(a) = \log_P(a)$, $h^{-1}(a) = P^a$

kde P je tajemství

h je homo z $(\mathbb{R},*)$ do $(\mathbb{R},+)$:

$h(a*b) = \log_P(a*b) = \log_P(a) + \log_P(b) = h(a) + h(b)$ ✓

$h^{-1}(h(a)+h(b)) = P^{\log_P(a*b)} = a*b$, t.j. funguje to

$g(a) = P^a$, $g^{-1}(a) = \log_P(a)$

g je homo z $(\mathbb{R},+)$ do $(\mathbb{R},*)$:

$g(a+b) = P^{(a+b)} = P^a * P^b = g(a) * g(b)$ ✓

$g^{-1}(g(a)*g(b)) = \log_P(P^{a+b}) = a+b$ ...

funguje to

Svazy

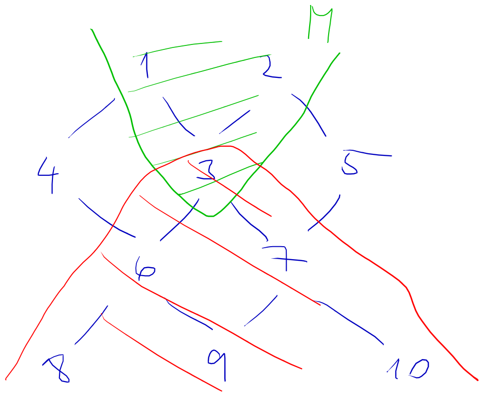
Pojmy: částečně uspořádaná množina, minimum, maximum, největší/nejmenší prvek, dolní/horní závora, supremum, infimum, Hasseův diagram, svaz

např.

$3 \cap 5 = 7$

$6 \cup 5 = 2$

$6 \cap 5 = 9$



$a < b \Leftrightarrow$ cesta z a nahoru do b

např. $10 < 1$

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

- nemá největší prvek
- maxima jsou 1 a 2
- nejmenší prvek = jediné minimum = 3
- dolní závora M je $\{3, 6, 7, 8, 9, 10\}$
- infimum je 3
- $\{1, 2\}$ má stejnou dolní závoru i infimum 3

Není svaz protože ne všechny páry / konečné podmnožiny mají sup. a inf., např. $8 \cap 9$ neexistuje

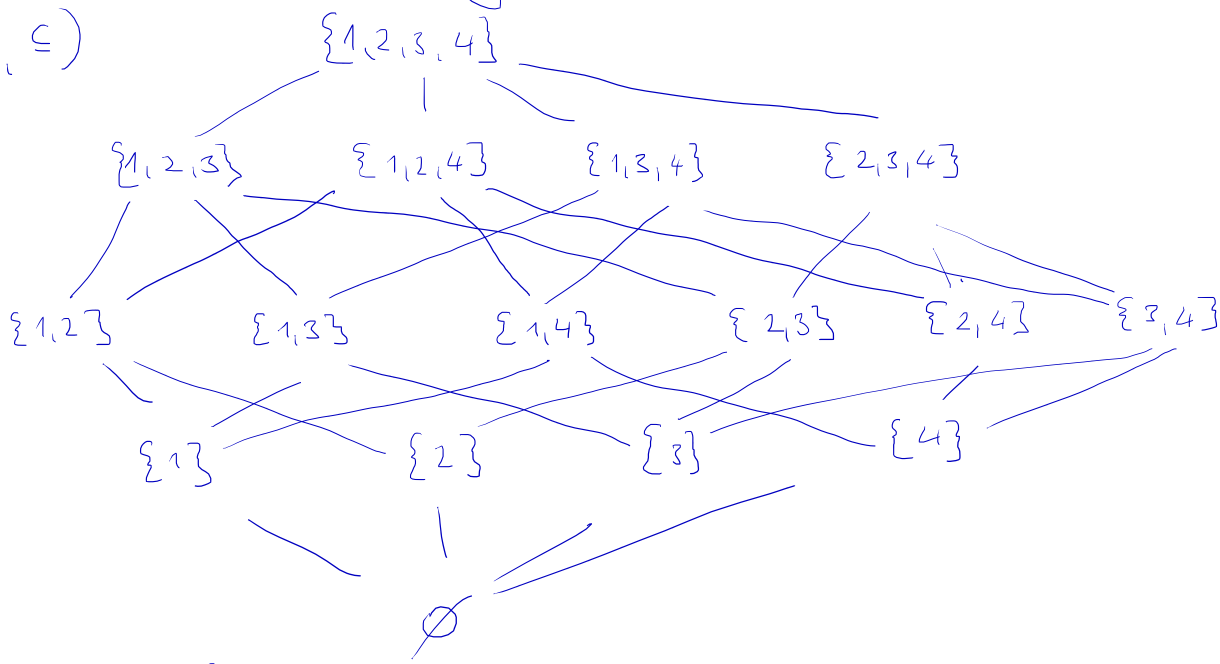
---

Úplný svaz: všechny podmnožiny (i nekonečné) mají sup. a inf.

- konečný svaz je vždy úplný
- $(\mathbb{N}, \leq)$ je svaz, není úplný protože $\mathbb{N}$ nemá sup.
- $(\mathbb{N} \cup \{\infty\}, \leq)$ je úplný svaz, $\sup(\mathbb{N}) = \infty$
- $(\mathbb{R}, \leq)$ svaz, ne úplný (není $\sup(\mathbb{R})$),
- $(\mathbb{R} \cup \{-\infty, \infty\}, \leq)$ úplný svaz
- $(\mathbb{Q} \cup \{-\infty, \infty\}, \leq)$ není úplný svaz, např. $\langle -\infty, \sqrt{2})$ nemá supremum, protože $\sqrt{2} \notin \mathbb{Q}$.

Graf podmnožin dané množiny

$(2^{\{1,2,3,4\}}, \subseteq)$



$\sup(\{\{1\},\{2\}\}) = \{1\} \cup \{2\} = \{1,2\}$

$\inf(\{\emptyset\}) = \emptyset$

$\inf(\emptyset) = \{1,2,3,4\}$, $\sup(\emptyset) = \emptyset$

Na svaz je možno nahlížet dvěma způsoby:

usp. množiny × algebry

$(2^{\{1,2,3,4\}}, \subseteq) \times (2^{\{1,2,3,4\}}, \cap, \cup)$

$\sup(\{a,b\}) = a \cup b$

$\inf(\{a,b\}) = a \cap b$

Definice 1 *Svaz* $(V, \leq)$ *je* úplný *pokud pro každou* $U \subseteq V$, $\inf(U) \in V$ *a* $\sup(U) \in V$.

Definice 2 *Mějme dva svazy* $(V, \leq)$ *a* $(V', \leq')$ *a funkci* $f : V \to V'$.

1. f *je* monotónní *pokud* $\forall u, v \in V : u \leq v \implies f(u) \leq' f(v)$.
2. f *je* spojitá *pokud je monotónní a pro každý (i nekonečný) řetěz C platí*

$$f(\sup(C)) = \sup\{f(c) \mid c \in C\} .$$

Věta 1 (Knaster-Tarski) *Mějme úplný svaz* $(V, \leq)$ *a monotónní funkci* $f : V \to V$. *Potom množina pevných bodů* f *je také úplným svazem.*

Zejména existuje *nejmenší pevný bod* μf a *největší pevný bod* νf.

Věta 2 (Kleene) *Nechť je* $(V, \leq)$ *úplný svaz a* $f : V \to V$ *spojitá funkce. Potom*

$$\mu f = \sup\{f^i(\perp) \mid i \leq 0\} .$$

μf může být vypočítán jako supremum neklesajícího řetězce

$$\perp \leq f(\perp) \leq f(f(\perp)) \leq f^3(\perp) \leq \cdots$$

Podobně nejmenší fixpoint větší než nějaký prvek x může být vypočítán jako supremum řetězce

$$x \leq f(x) \leq f(f(x)) \leq f^3(x) \leq \cdots$$

Duálně pro νf.

Věta 3 (slabší Knaster-Tarski) *Nechť* $(V, \leq)$ *je částečně uspořádaná množina s nejmenším prvkem* $\perp$, *kde každý nekonečný řetěz má supremum, a nechť je* $f : V \to V$ *spojitá funkce. Pak* $\mu f = \sup^{i \geq 0} f^i(\perp)$ *je nejmenším pevným bodem* f.

Př: $V = 2^{\mathbb{N}}$

Svaz $(V, \subseteq)$

$f : V \to V$

$X \mapsto X \cup \{0, \dots, \min(|X|, 3)\}$

Je f monotónní?

$$A \subseteq B \overset{?}{\Rightarrow} f(A) \subseteq f(B)$$

$$A \cup \{0, \dots, \min(|A|,3)\} \overset{?}{\subseteq} B \cup \{0, \dots, \min(|B|,3)\}$$

platí, protože

$$A \subseteq B \Rightarrow \min(|A|,3) \le \min(|B|,3) \Rightarrow$$
$$\{0, \dots, \min(|A|,3)\} \subseteq \{0, \dots, \min(|B|,3)\}$$

f je monotónní ✓

f je i spojitá (DÚ)

Platí Kleene i Knaster-Tarski

- pevné body f tvoří úplný svaz
- μf můžu počítat jako supremum $f^0(\bot), f^1(\bot), \dots$

$f \circlearrowleft \{0,1,2,3\} = \mu f$

$\uparrow f$

$\{0,1,2\}$

$\uparrow f$

$\{0,1\}$

$\uparrow f$

$\{0\}$

$\uparrow f$

$\emptyset$

Kleene iterace spočítá μf.

$\sqcup$ $\sqcup$

fixpointy f jsou svaz

$g : V \to V$

$X \mapsto X \cup \{\min(|X|, 3)\}$

$g \circlearrowleft \{0,1,2,3\}$ není μf!

$g \uparrow$

$\{0,1,2\}$

$g \uparrow$

$\{0,1\}$

$g \uparrow$

$\{0\}$

$g \uparrow$

$\emptyset$

(např. $\{0,2\}$ je menší)

Kleene iterace nefunguje.

pevné body g nejsou svaz: např. $\{0,2\} \cup \{1\}$ není pevný bod

→ protože g není monotónní

$$A \subseteq B \not\Rightarrow g(A) \subseteq g(B)$$

např. pro $A = \emptyset$, $B = \{1\}$

Kde člověk potká výpočet μf Kleeneho iterací

formulace algoritmů

* Opora TIN algo 4.2., výpočet dosažitelných symbolů

$$f(M) = \{S\} \cup \{X \mid A \to \alpha X \beta \in P \wedge A \in M\}$$

dostupné symboly $V = \mu f$

$f^0(\emptyset) = \emptyset$

$f^1(\emptyset) = S$

$f^2(\emptyset) = f(\{S\}) =$ symb na pravé straně S pravidla

$f^3(\emptyset)$ ....

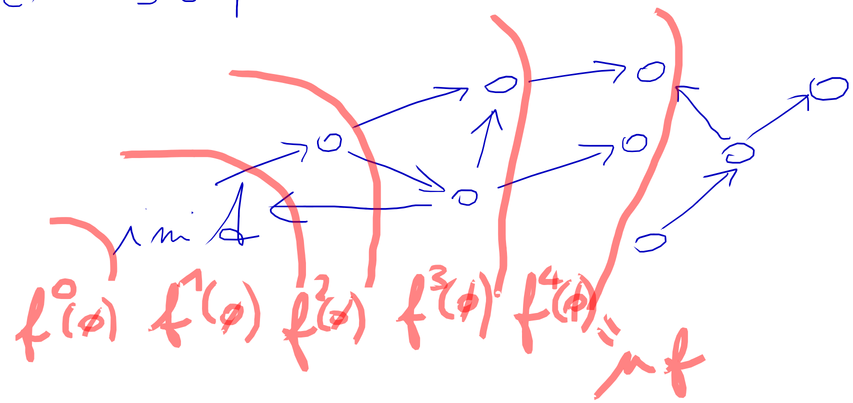
$\vdots$

Podobně lze fixpoint notaci použít např. v alg 4.1, 4.4., a v mnoha dalších.

* Které uzly jsou dosažitelné v grafu z vrcholu init?

$f(X) = \{init\} \cup$ přímí následníci vrcholů v X

μf



Denotační sémantika programů

- formální definice významu zdrojového kódu, důležité např. ve verifikaci (existuje více přístupů)

p je program (kód), jeho sémantika $[\![p]\!]: State \to State$ je funkce modifikující stav proměnných.

$[\![x := x+1]\!]$

$0 \mapsto 1$
$1 \mapsto 2$
$2 \mapsto 3$
$\vdots$

$[\![x := x \% 2]\!]$

$0 \mapsto 0$
$1 \mapsto 1$
$2 \mapsto 0$
$3 \mapsto 1$
$\vdots$

$[\![p ; p']\!] = p' \circ p$

např. $[\![x := x+1 ; x := x \% 2]\!] = [\![x := x \% 2]\!] \circ [\![x := x+1]\!]$

$0 \mapsto 1$
$1 \mapsto 0$
$2 \mapsto 1$
$3 \mapsto 0$

$$[\![\text{if } x=0 \text{ then } a \text{ else } b]\!] = \{(x,x') \in [\![a]\!] \mid x = 0\} \cup \{(x,x') \in [\![b]\!] \mid x \neq 0\}$$

Zapeklité jsou cykly : $[\![\text{while test do body}]\!]$

Nechť f^i je sémantika cyklu pokud skončí po méně než i iteracích.

Potom
$$f^{i+1} = \{(x,x) \mid x \text{ nesplňuje test}\}$$
$$\cup$$
$$\{(x,x') \mid x \text{ splňuje test a } (x,x') \in f^i \circ [\![\text{body}]\!]\}$$

To je předpis, jak z f^i vyrobit f^{i+1}, funkce $\Gamma : (State \to State) \to (State \to State)$.

Sémantika se pak definuje jednoduše jako

$$[\![\text{while test do body}]\!] = \mu\Gamma$$

Např.
$$\Gamma^0(\emptyset) = \emptyset$$
$$\Gamma^1(\emptyset) = \{(x,x) \mid x > 1\}$$
$$\Gamma^2(\emptyset) = \Gamma^1(\emptyset) \cup \{(1,2)\}$$
$$\Gamma^3(\emptyset) = \Gamma^2(\emptyset) \cup \{(0,2)\}$$
$$\Gamma^4(\emptyset) = \Gamma^3(\emptyset) \cup \{(-1,2)\}$$
$$\vdots$$
$$\Gamma^{i+1}(\emptyset) = \Gamma^i(\emptyset) \cup \{(2-i,2)\}$$
$$\vdots$$
$$\mu\Gamma = \{(2-i,2) \mid i \in \mathbb{N}\}$$