

Linguaggi di Programmazione

Roberta Gori

Esercitazione #3

Ricorsione ben fondata

Definire usando la ricorsione ben fondata una funzione $vars$ tale che data un' espressione aritmetica a ritorna l'insieme degli identificatori che appaiono nell'espressione aritmetica a . Poi provare per induzione sulle regole che

$$\forall a \in Aexp, \forall \sigma \in \Sigma, \forall n \in \mathbb{Z}$$

$$\langle a, \sigma \rangle \rightarrow n \quad \text{implica} \quad (\forall y \in vars(a). \sigma(y) = \sigma'(y)) \Rightarrow \langle a, \sigma' \rangle \rightarrow n).$$

se due memorie coincidono su tutte le variabili che appaiono in un'espressione, allora valutando l'espressione nelle due memorie dà lo stesso risultato

Ex. 1, Vars

$$vars : Aexp \rightarrow \wp(Var)$$

$$\begin{aligned} vars(n) &\triangleq \emptyset \\ vars(x) &\triangleq \{x\} \\ vars(a_1 \text{ op } a_2) &\triangleq vars(a_1) \cup vars(a_2) \end{aligned}$$

(ricorsione ben fondata:
relazione immediata di sotto termine)

Ex. 1, Induzione

$$P(\langle a, \sigma \rangle \rightarrow n) \triangleq \forall \sigma'. (\forall y \in \text{vars}(a). \sigma'(y) = \sigma(y)) \Rightarrow \langle a, \sigma' \rangle \rightarrow n$$

$$\frac{}{\langle n, \sigma \rangle \rightarrow n} \text{ (num)}$$

$$P(\langle n, \sigma \rangle \rightarrow n) \triangleq \forall \sigma'. \boxed{(\forall y \in \boxed{\text{vars}(n)}. \sigma'(y) = \sigma(y))} \Rightarrow \langle n, \sigma' \rangle \rightarrow n$$

$\emptyset$

tt

per regola (num) $\langle n, \sigma' \rangle \rightarrow n$

Ex. 1, Induzione

$$P(\langle a, \sigma \rangle \rightarrow n) \triangleq \forall \sigma'. (\forall y \in vars(a). \sigma'(y) = \sigma(y)) \Rightarrow \langle a, \sigma' \rangle \rightarrow n$$

$$\frac{}{\langle x, \sigma \rangle \rightarrow \sigma(x)} \text{(ide)}$$

$$P(\langle x, \sigma \rangle \rightarrow \sigma(x)) \triangleq \forall \sigma' \left[\begin{array}{c} (\forall y \in \boxed{vars(x)} \cdot \sigma'(y) = \sigma(y)) \\ \{x\} \end{array} \Rightarrow \langle x, \sigma' \rangle \rightarrow \sigma(x) \right]$$
$$\sigma'(x) = \sigma(x)$$

Assumiamo

$$\sigma'(x) = \sigma(x)$$

per regola (ide) $\langle x, \sigma' \rangle \rightarrow \sigma'(x) = \sigma(x)$

Ex. 1, Induzione

$$\frac{\langle a_1, \sigma \rangle \rightarrow n_1 \quad \langle a_2, \sigma \rangle \rightarrow n_2}{\langle a_1 \text{ op } a_2, \sigma \rangle \rightarrow n_1 \text{ op } n_2}$$

Assumiamo

$$P(\langle a_i, \sigma \rangle \rightarrow n_i) \triangleq \forall \sigma'. (\forall y \in \text{vars}(a_i). \sigma'(y) = \sigma(y)) \Rightarrow \langle a_i, \sigma' \rangle \rightarrow n_i$$

Vogliamo provare

$$P(\langle a_1 \text{ op } a_2, \sigma \rangle \rightarrow n_1 \text{ op } n_2) \triangleq \forall \sigma'.$$

$$(\forall y \in \text{vars}(a_1 \text{ op } a_2). \sigma'(y) = \sigma(y)) \Rightarrow \langle a_1 \text{ op } a_2, \sigma' \rangle \rightarrow n_1 \text{ op } n_2$$

Assumiamo

$$\forall y \in \boxed{\text{vars}(a_1 \text{ op } a_2)} \quad \sigma'(y) = \sigma(y)$$

$$(\forall y \in \text{vars}(a_1). \sigma'(y) = \sigma(y)) \wedge (\forall y \in \text{vars}(a_2). \sigma'(y) = \sigma(y))$$

per ipotesi induttiva

$$\langle a_1, \sigma' \rangle \rightarrow n_1 \quad \text{E} \quad \langle a_2, \sigma' \rangle \rightarrow n_2$$

$$\text{per regola (op)} \quad \langle a_1 \text{ op } a_2, \sigma' \rangle \rightarrow n_1 \text{ op } n_2$$

[Ex. 2] Definire usando la ricorsione ben fondata una funzione *vars* tale che data un comando *c* ritorna l'insieme degli identificatori che appaiono a sinistra degli assegnamenti.

Poi provare per induzione sulle regole che

$$\langle c, \sigma \rangle \rightarrow \sigma' \quad \text{implica} \quad \forall x \notin \text{vars}(c). \sigma(x) = \sigma'(x).$$

se una variabile non appare in un'assegnamento allora il suo valore iniziale viene conservato nello store finale

Ex. 2, Vars

$$vars : Com \rightarrow \wp(Ide)$$

$$\begin{aligned} vars(\mathbf{skip}) &\triangleq \emptyset \\ vars(x := a) &\triangleq \{x\} \\ vars(c_1; c_2) &\triangleq vars(c_1) \cup vars(c_2) \\ vars(\mathbf{if } b \mathbf{ then } c_1 \mathbf{ else } c_2) &\triangleq vars(c_1) \cup vars(c_2) \\ vars(\mathbf{while } b \mathbf{ do } c) &\triangleq vars(c) \end{aligned}$$

ricorsione ben fondata:

relazione immediata di sotto termine

Ex. 2, Induzione

$$P(\langle c, \sigma \rangle \rightarrow \sigma') \triangleq \forall y \notin \text{vars}(c). \sigma'(y) = \sigma(y)$$

$$\frac{}{\langle \mathbf{skip}, \sigma \rangle \rightarrow \sigma}$$

Vogliamo provare

$$P(\langle \mathbf{skip}, \sigma \rangle \rightarrow \sigma) \triangleq \forall y \notin \boxed{\text{vars}(\mathbf{skip})} \cdot \sigma(y) = \sigma(y)$$

$\emptyset$

$$\forall y. \sigma(y) = \sigma(y)$$

ovvio

Ex. 2, Induction

$$P(\langle c, \sigma \rangle \rightarrow \sigma') \triangleq \forall y \notin \text{vars}(c). \sigma'(y) = \sigma(y)$$

$$\frac{\langle a, \sigma \rangle \rightarrow n}{\langle x := a, \sigma \rangle \rightarrow \sigma[n/x]}$$

Vogliamo provare

$$P(\langle x := a, \sigma \rangle \rightarrow \sigma[n/x]) \triangleq \forall y \notin \boxed{\text{vars}(x := a)} \cdot \sigma[n/x](y) = \sigma(y)$$

$\{x\}$

$$\forall y \neq x. \sigma[n/x](y) = \sigma(y)$$

per def di $\sigma[n/x]$

Ex. 2

$$\frac{\langle c_1, \sigma \rangle \rightarrow \sigma'' \quad \langle c_2, \sigma'' \rangle \rightarrow \sigma'}{\langle c_1; c_2, \sigma \rangle \rightarrow \sigma'}$$

Assumiamo

$$P(\langle c_1, \sigma \rangle \rightarrow \sigma'') \triangleq \forall y \notin \text{vars}(c_1). \sigma''(y) = \sigma(y)$$

$$P(\langle c_2, \sigma'' \rangle \rightarrow \sigma') \triangleq \forall y \notin \text{vars}(c_2). \sigma'(y) = \sigma''(y)$$

Vogliamo provare

$$P(\langle c_1; c_2, \sigma \rangle \rightarrow \sigma') \triangleq \forall y \notin \boxed{\text{vars}(c_1; c_2)}. \sigma'(y) = \sigma(y)$$

$\text{vars}(c_1) \cup \text{vars}(c_2)$

Prendiamo

$$\forall y \notin \text{vars}(c_1) \cup \text{vars}(c_2). \sigma'(y) = \sigma(y)$$

$$y \notin \text{vars}(c_1) \cup \text{vars}(c_2)$$

Dal momento che

$$y \notin \text{vars}(c_2) \text{ allora per ip. ind. } \sigma'(y) = \sigma''(y)$$

Dal momento che

$$y \notin \text{vars}(c_1) \text{ allora per ip. ind. } \sigma''(y) = \sigma(y)$$

$$\text{Quindi } \sigma'(y) = \sigma(y)$$

Ex. 2

$$\frac{\langle b, \sigma \rangle \rightarrow \mathbf{tt} \quad \langle c_1, \sigma \rangle \rightarrow \sigma'}{\langle \mathbf{if } b \mathbf{ then } c_1 \mathbf{ else } c_2, \sigma \rangle \rightarrow \sigma'}$$

Assumiamo

$$P(\langle c_1, \sigma \rangle \rightarrow \sigma') \triangleq \forall y \notin \mathit{vars}(c_1). \sigma'(y) = \sigma(y)$$

Vogliamo provare

$$P(\langle \mathbf{if } b \mathbf{ then } c_1 \mathbf{ else } c_2, \sigma \rangle \rightarrow \sigma') \triangleq \forall y \notin \boxed{\begin{array}{c} \mathit{vars}(\mathbf{if } b \mathbf{ then } c_1 \mathbf{ else } c_2) \\ \mathit{vars}(c_1) \cup \mathit{vars}(c_2) \end{array}} \sigma'(y) = \sigma(y)$$
$$\forall y \notin \mathit{vars}(c_1) \cup \mathit{vars}(c_2). \sigma'(y) = \sigma(y)$$

Prendiamo

$$y \notin \mathit{vars}(c_1) \cup \mathit{vars}(c_2)$$

Dal momento che

$$y \notin \mathit{vars}(c_1) \quad \text{allora per ip. ind.} \quad \sigma'(y) = \sigma(y)$$

Ex. 2

$$\frac{\langle b, \sigma \rangle \rightarrow \mathbf{false}}{\langle \mathbf{while } b \mathbf{ do } c, \sigma \rangle \rightarrow \sigma}$$

Vogliamo provare

$$P(\langle \mathbf{while } b \mathbf{ do } c, \sigma \rangle \rightarrow \sigma) \triangleq \forall y \notin \boxed{\begin{array}{c} vars(\mathbf{while } b \mathbf{ do } c) \\ vars(c) \end{array}}. \sigma(y) = \sigma(y)$$

$$\forall y \notin vars(c). \sigma(y) = \sigma(y)$$

ovvio

Ex. 2

$$\frac{\langle b, \sigma \rangle \rightarrow \mathbf{true} \quad \langle c, \sigma \rangle \rightarrow \sigma'' \quad \langle \mathbf{while} \ b \ \mathbf{do} \ c, \sigma'' \rangle \rightarrow \sigma'}{\langle \mathbf{while} \ b \ \mathbf{do} \ c, \sigma \rangle \rightarrow \sigma'}$$

Assumiamo

$$P(\langle c, \sigma \rangle \rightarrow \sigma'') \triangleq \forall y \notin vars(c). \sigma''(y) = \sigma(y)$$

$$P(\langle \mathbf{while} \ b \ \mathbf{do} \ c, \sigma'' \rangle \rightarrow \sigma') \triangleq \forall y \notin \boxed{vars(\mathbf{while} \ b \ \mathbf{do} \ c)} \sigma'(y) = \sigma''(y)$$

$vars(c)$

Vogliamo provare

$$P(\langle \mathbf{while} \ b \ \mathbf{do} \ c, \sigma \rangle \rightarrow \sigma') \triangleq \forall y \notin \boxed{vars(\mathbf{while} \ b \ \mathbf{do} \ c)} \sigma'(y) = \sigma(y)$$

$vars(c)$

Prendiamo

$$y \notin vars(c)$$

Dal momento che

$$y \notin vars(c)$$

Dal momento che

$$y \notin vars(c)$$

$$\text{allora per ip. ind. } \sigma'(y) = \sigma''(y)$$

$$\text{allora per ip. ind. } \sigma''(y) = \sigma(y)$$

$$\text{Quindi } \sigma'(y) = \sigma(y)$$

Funzioni monotone e continue

[Ex. 3] Consideriamo l' $\text{OPC}_\perp (\wp(\mathbb{N}), \subseteq)$. Proviamo che per ogni set $S \subseteq \mathbb{N}$:

1. la funzione $f_S : \wp(\mathbb{N}) \rightarrow \wp(\mathbb{N})$ tale che $f_S(X) = X \cap S$ e' continua
2. la funzione $g_S : \wp(\mathbb{N}) \rightarrow \wp(\mathbb{N})$ tale che $g_S(X) = X \cup S$ e' continua

omettiamo di controllare la monotonicit 

Ex. 3, Continuità'

Prendiamo una catena

$$\{X_i\}_{i \in \mathbb{N}}$$

$$f_S(X) = X \cap S$$

Vogliamo provare $f_S \left(\bigcup_{i \in \mathbb{N}} X_i \right) = \bigcup_{i \in \mathbb{N}} f_S(X_i)$

$$f_S \left(\bigcup_{i \in \mathbb{N}} X_i \right) = \left(\bigcup_{i \in \mathbb{N}} X_i \right) \cap S = \bigcup_{i \in \mathbb{N}} (X_i \cap S) = \bigcup_{i \in \mathbb{N}} f_S(X_i)$$

per def per distributivita' per def

Ex. 3, Continuity

Prendiamo una catena $g_S(X) = X \cup S$
 $\{X_i\}_{i \in \mathbb{N}}$

Vogliamo provare $g_S \left(\bigcup_{i \in \mathbb{N}} X_i \right) = \bigcup_{i \in \mathbb{N}} g_S(X_i)$

$$g_S \left(\bigcup_{i \in \mathbb{N}} X_i \right) = \left(\bigcup_{i \in \mathbb{N}} X_i \right) \cup S = \bigcup_{i \in \mathbb{N}} (X_i \cup S) = \bigcup_{i \in \mathbb{N}} g_S(X_i)$$

def def per idempotenza per def

[Ex. 4] Provare che ogni funzione che preserva il limite e' monotona.

Ex. 4, Funzioni che preservano il limite

$(D, \sqsubseteq_D)$ CPO $(E, \sqsubseteq_E)$ CPO $f : D \rightarrow E$ preservano il limite

$$f \left(\bigsqcup_{i \in \mathbb{N}} d_i \right) = \bigsqcup_{i \in \mathbb{N}} f(d_i)$$

vogliamo provare la monotonicit 

prendiamo $d \sqsubseteq_D d'$

vogliamo provare $f(d) \sqsubseteq_E f(d')$

consideriamo $d_i = \begin{cases} d & \text{if } i = 0 \\ d' & \text{otherwise} \end{cases}$ quindi $\bigsqcup_{i \in \mathbb{N}} d_i = d'$

allora

$$\bigsqcup_{i \in \mathbb{N}} f(d_i) = f \left(\bigsqcup_{i \in \mathbb{N}} d_i \right) = f(d')$$

$$\text{e } f(d) = f(d_0) \sqsubseteq_E \bigsqcup_{i \in \mathbb{N}} f(d_i) = f(d')$$

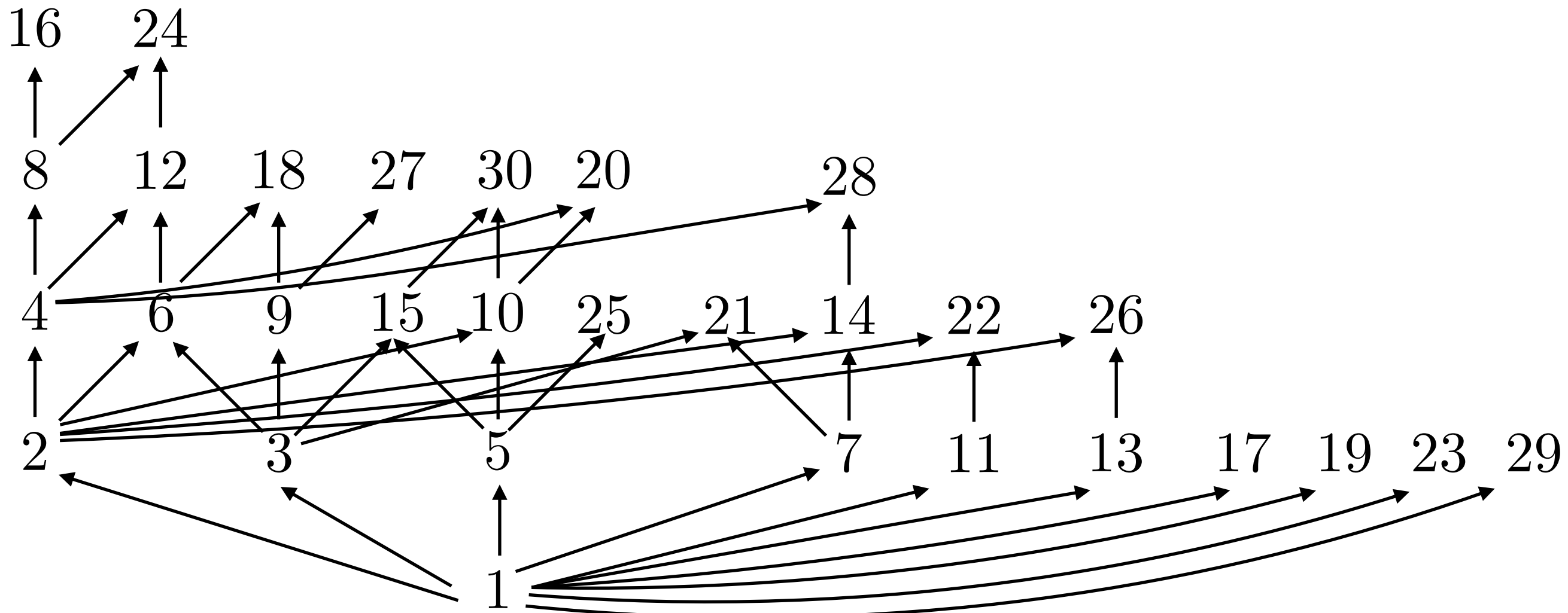
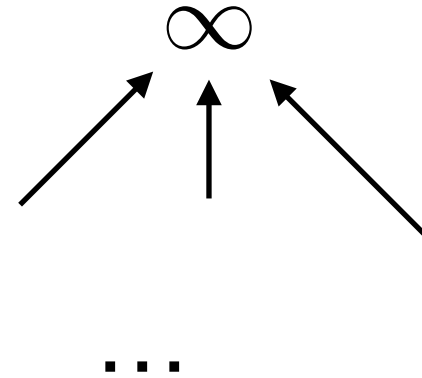
OPC

[Ex. 5] Sia $D = \{n \in \mathbb{N} \mid n > 0\} \cup \{\infty\}$ e $\sqsubseteq \subseteq (D \times D)$ tale che

- per ogni $n, m \in D \cap \mathbb{N}$, definiamo $n \sqsubseteq m$ se n divide m ;
- per ogni $x \in D$, definiamo $x \sqsubseteq \infty$.

E' $(D, \sqsubseteq)$ un OPC $_{\perp}$?

Ex. 5, divide



Ex. 5, divide

$OPC_{\perp}$?

riflessivo?

$$d \in D \begin{cases} d \in \mathbb{N} \\ d = \infty \end{cases}$$

d divide d

$$d \sqsubseteq d$$

$$\infty \sqsubseteq \infty$$

antisimmetrico?

$$d_1, d_2 \in D \quad \begin{matrix} d_1 \sqsubseteq d_2 \\ d_2 \sqsubseteq d_1 \end{matrix}$$

$$d_1 \stackrel{?}{=} d_2$$

$$d_1, d_2 \in \mathbb{N}$$

$$\begin{matrix} d_1 \text{ divide } d_2 \\ d_2 \text{ divide } d_1 \end{matrix}$$

$$d_1 \leq d_2$$

$$d_2 \leq d_1$$

$$d_1 = d_2$$

$$d_1 = \infty$$

$$\infty \sqsubseteq d_2$$

$$d_2 = \infty$$

$$d_2 = \infty = d_1$$

$$d_2 = \infty$$

$$\infty \sqsubseteq d_1$$

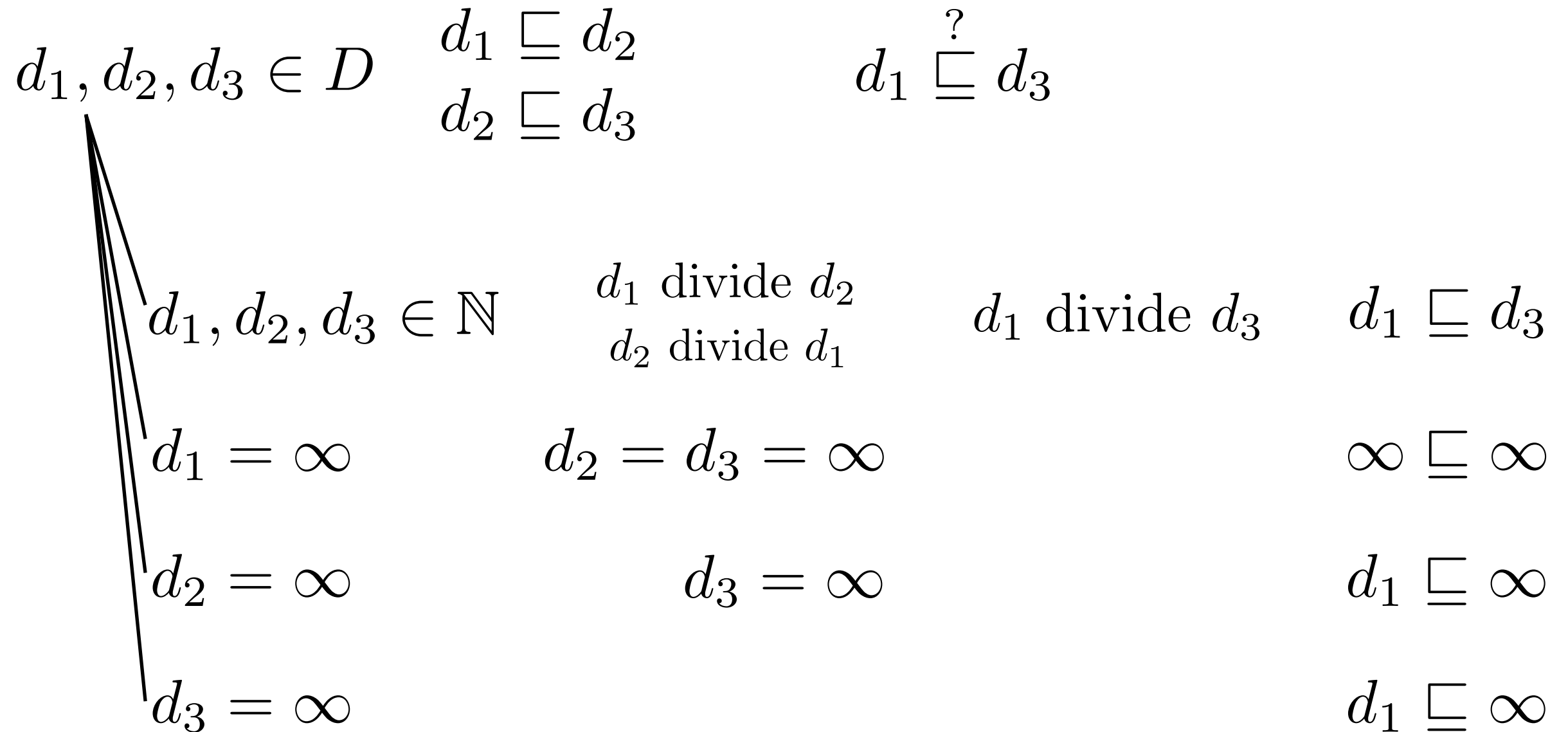
$$d_1 = \infty$$

$$d_1 = \infty = d_2$$

Ex. 5, divides

$OPC_{\perp}$?

transitivo?



bottom? 1 divide ogni numero e $1 \sqsubseteq \infty$

Ex. 5, divides

$OPC_{\perp}$?

completo?

ogni catena finita ha un limite

catene infinite possono contenere solo numeri sempre piu' grandi, allora il limite e' ∞

Punti fissi

[**Ex. 6**] Definiamo due funzioni $f_i : D_i \rightarrow D_i$ su due OPC D_i per $i \in \{1, 2\}$ (non necessariamente con bottom) tali che:

1. f_1 è continua, ha punti fissi ma non minimo punto fisso;
2. f_2 è continua e non ha punti fissi.

Ex. 6, punti fissi

1. f_1 è continua, ha punti fissi ma non minimo punto fisso;

Troviamo un esempio minimale

Di quanti elementi abbiamo bisogno come minimo?

Come dovrebbero essere ordinati?

Ex. 6, punti fissi

1. f_1 è continua, ha punti fissi ma non minimo punto fisso;

$$D_1 = (\{0, 1\}, =)$$

$$f_1 : D_1 \rightarrow D_1$$

OPC

finito: ogni funzione monotona e' continua

$$f_1(0) = 0$$

$$f_1(1) = 1$$

il th di Kleene non e' applicabile: perche'?

Ex. 6, punti fissi

2. f_2 è continua e non ha punti fissi;

Troviamo un esempio minimale

Di quanti elementi abbiamo bisogno come minimo?

Come dovrebbero essere ordinati?

Ex. 6, punti fissi

2. f_2 è continua e non ha punti fissi;

$$D_2 = D_1 = (\{0, 1\}, =) \quad f_2 : D_2 \rightarrow D_2$$

OPC

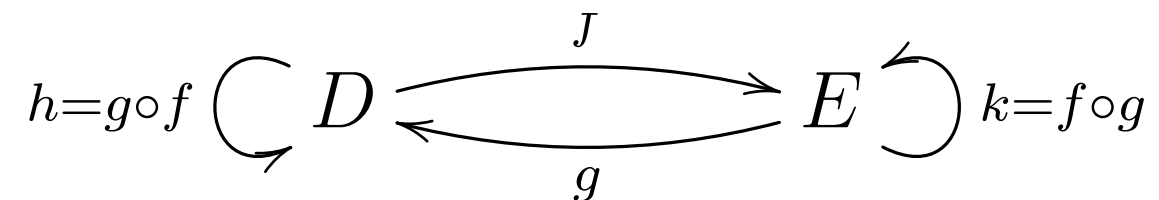
finito: ogni funzione monotona e' continua

$$f_2(0) = 1$$

$$f_2(1) = 0$$

il th di Kleene non e' applicabile: perche'?

[Ex.7] Siano D, E due $OPC_{\perp}$ e siano $f : D \rightarrow E$ e $g : E \rightarrow D$ due funzioni continue. La loro composizione $h = g \circ f : D \rightarrow D$ e $k = f \circ g : E \rightarrow E$ sono continue e per questo hanno punti fissi.



Sia $e_0 = fix(k) \in E$. Provare che $g(e_0) = fix(h) \in D$ mostrando che

1. $g(e_0)$ è un punto fisso per h ;
2. $g(e_0)$ è il minimo pre-punto fisso per h .

Ex. 7, composizione

$$h=g \circ f \quad \begin{array}{c} \curvearrowright \\ D \end{array} \begin{array}{c} \xrightarrow{f} \\ \xleftarrow{g} \end{array} E \begin{array}{c} \curvearrowleft \\ \end{array} k=f \circ g \quad e_0 = \text{fix}(k)$$

1. $g(e_0)$ è un punto fisso per h ;

dobbiamo provare $h(g(e_0)) = g(e_0)$

$$h(g(e_0)) = g(f(g(e_0))) = g(k(e_0)) = g(e_0)$$

per def

per def $e_0 = \text{fix}(k)$

Ex. 7, composizione

$$h=g \circ f \quad \left(\curvearrowright \right) D \begin{array}{c} \xrightarrow{f} \\ \xleftarrow{g} \end{array} E \begin{array}{c} \curvearrowleft \\ \curvearrowright \end{array} k=f \circ g \quad e_0 = \text{fix}(k)$$

2. $g(e_0)$ è il minimo pre-puntofisso per h ;

prendiamo

$$d \sqsupseteq_D h(d) \quad \text{vogliamo provare} \quad g(e_0) \sqsubseteq_D d$$

$$d \sqsupseteq_D h(d) = g(f(d)) \quad (\text{by def of } h)$$

$$f(d) \sqsupseteq_E f(g(f(d))) = k(f(d)) \quad (\text{per monotonicita' di } f, \text{ def. } k)$$

$$\left. \begin{array}{l} \text{percio' } f(d) \text{ e' un pre-puntofisso of } k \\ e_0 \text{ e' il minimo pre-puntofisso of } k \end{array} \right\} \Rightarrow e_0 \sqsubseteq_E f(d)$$

$$g(e_0) \sqsubseteq_D g(f(d)) = h(d) \sqsubseteq_D d \quad (\text{per mon. } g, \text{ def. } h, \text{ ipo. } d)$$