PSC 2020/21 (375AA, 9CFU)

Principles for Software Composition

Roberto Bruni
http://www.di.unipi.it/~bruni/

http://didawiki.di.unipi.it/doku.php/magistraleinformatica/psc/

Exercises #3

Well-founded recursion

[**Ex. 1**] Define by well-founded recursion the function *vars* that, given an arithmetic expression a, returns the set of identifiers that appear in a. Then, prove by rule induction that $\forall a \in Aexp, \forall \sigma \in \Sigma, \forall n \in \mathbb{Z}$

$$\langle a, \sigma \rangle \to n \qquad \text{implies} \qquad \forall \sigma'. \, (\, (\forall y \in vars(a). \, \sigma(y) = \sigma'(y)) \Rightarrow \langle a, \sigma' \rangle \to n \,).$$

if two memories coincide on all variables that appear in one expression, then evaluating the expression in the two memories give the same result

Ex. 1, Vars

$$vars : Aexp \to \wp(Ide)$$

$$\begin{array}{rcl} vars(n) & \triangleq & \emptyset \\ vars(x) & \triangleq & \{x\} \\ vars(a_1 \ \mathsf{op} \ a_2) & \triangleq & vars(a_1) \cup vars(a_2) \end{array}$$

(well founded recursion by immediate subterm relation)

Ex. 1, Induction

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

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

$$P(\langle n, \sigma \rangle \to n) \triangleq \forall \sigma'.\ \boxed{(\forall y \in \underbrace{\boxed{vars(n)}}_{\emptyset}.\ \sigma'(y) = \sigma(y))}_{\mathbf{tt}} \Rightarrow \langle n, \sigma' \rangle \to n$$

by (num) $\langle n, \sigma' \rangle \to n$

Ex. 1, Induction

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

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

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

$$\{x\}$$

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

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

by (ide) $\langle x, \sigma' \rangle \to \sigma'(x) = \sigma(x)$

Ex. 1, Induction

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

Assume

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

We want to prove

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

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

$vars(a_1) \cup vars(a_2)$

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

by inductive hypotheses

$$\langle a_1, \sigma'\rangle \to n_1 \qquad \langle a_2, \sigma'\rangle \to n_2$$

by (op) $\langle a_1 \text{ op } a_2, \sigma'\rangle \to n_1 \ op \ n_2$

[**Ex. 2**] Define by well-founded recursion the function *vars* that, given a command, returns the set of identifiers that appear on the left-hand side of some assignment. Then, prove by rule induction that $\forall c \in Com, \forall \sigma, \sigma' \in \Sigma$

$$\langle c, \sigma \rangle \to \sigma' \qquad \text{implies} \qquad \forall x \notin vars(c).\ \sigma(x) = \sigma'(x).$$

if a variable does not appear in an assignment then its initial value is preserved in the final store

Ex. 2, Vars

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

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

(well founded recursion by immediate subterm relation)

Ex. 2, Induction

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

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

We want to prove

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

$$\emptyset$$

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

obvious

Ex. 2, Induction

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

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

We want to prove

$$P(\langle x := a, \sigma \rangle \rightarrow \sigma[^n/_x]) \triangleq \boxed{\forall y \notin \underbrace{\boxed{vars(x := a)}}_{\{x\}}.\ \sigma[^n/_x](y) = \sigma(y)}$$

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

by def of $\sigma[^n/_x]$

Ex. 2

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

Assume $P(\langle c_1, \sigma \rangle \to \sigma'') \triangleq \forall y \notin vars(c_1).\ \sigma''(y) = \sigma(y)$

$P(\langle c_2, \sigma'' \rangle \to \sigma') \triangleq \forall y \notin vars(c_2).\ \sigma'(y) = \sigma''(y)$

We want to prove

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

$vars(c_1) \cup vars(c_2)$

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

Take $y \notin vars(c_1) \cup vars(c_2)$

Since $y \notin vars(c_2)$ then by ind. hyp. $\sigma'(y) = \sigma''(y)$

Since $y \notin vars(c_1)$ then by ind. hyp. $\sigma''(y) = \sigma(y)$

Thus $\sigma'(y) = \sigma(y)$

Ex. 2

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

Assume $P(\langle c_1, \sigma\rangle \to \sigma') \triangleq \forall y \notin vars(c_1).\ \sigma'(y) = \sigma(y)$

We want to prove

$$P(\langle \mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2, \sigma\rangle \to \sigma') \triangleq \forall y \notin vars(\mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2).\ \sigma'(y) = \sigma(y)$$

$vars(c_1) \cup vars(c_2)$

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

Take $y \notin vars(c_1) \cup vars(c_2)$

Since $y \notin vars(c_1)$ then by ind. hyp. $\sigma'(y) = \sigma(y)$

Ex. 2

$$\frac{\langle b, \sigma \rangle \to \textbf{false}}{\langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma}$$

We want to prove

$$P(\langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma) \triangleq \forall y \notin vars(\textbf{while } b \textbf{ do } c).\ \sigma(y) = \sigma(y)$$

$$vars(c)$$

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

obvious

Ex. 2

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

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

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

$vars(c)$

We want to prove

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

$vars(c)$

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

Take $y \notin vars(c)$

Since $y \notin vars(c)$ then by ind. hyp. $\sigma'(y) = \sigma''(y)$

Since $y \notin vars(c)$ then by ind. hyp. $\sigma''(y) = \sigma(y)$

Thus $\sigma'(y) = \sigma(y)$

Monotone and continuous functions

[Ex. 3] Consider the CPO$_\perp$ $(\wp(\mathbb{N}), \subseteq)$. Prove that for any set $S \subseteq \mathbb{N}$:

1. the function $f_S : \wp(\mathbb{N}) \to \wp(\mathbb{N})$ such that $f_S(X) = X \cap S$ is continuous.
2. the function $g_S : \wp(\mathbb{N}) \to \wp(\mathbb{N})$ such that $g_S(X) = X \cup S$ is continuous.

we omit to check monotonicity

Ex. 3, Continuity

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

Take a chain $\{X_i\}_{i \in \mathbb{N}}$

We need to prove $f_S\left(\bigcup_{i \in \mathbb{N}} X_i\right) = \bigcup_{i \in \mathbb{N}} f_S(X_i)$

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

by def

by distributivity

by def

Ex. 3, Continuity

$$g_S(X) = X \cup S$$

Take a chain $\{X_i\}_{i \in \mathbb{N}}$

We need to prove $g_S\left(\bigcup_{i \in \mathbb{N}} X_i\right) = \bigcup_{i \in \mathbb{N}} g_S(X_i)$

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

by def by idempotency by def

[**Ex. 4**] Prove that any limit-preserving function is monotone.

Ex. 4, limit preserving

$(D, \sqsubseteq_D)$ CPO $\quad (E, \sqsubseteq_E)$ CPO $\quad f : D \to E$ limit-preserving

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

we want to prove monotonicity

take $d \sqsubseteq_D d'$

we want to prove $f(d) \sqsubseteq_E f(d')$

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

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

and $f(d) = f(d_0) \sqsubseteq_E f(d')$

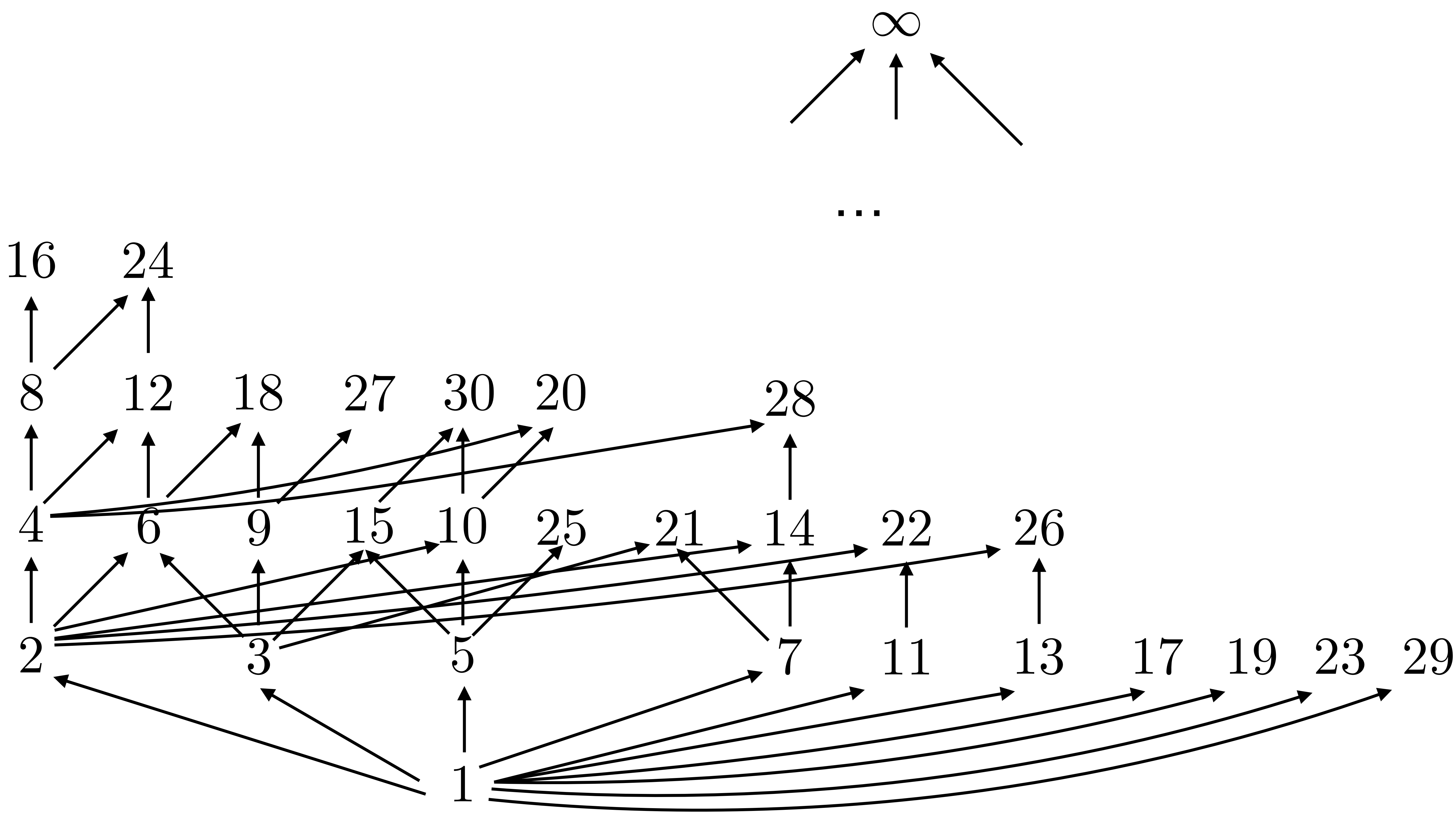
CPOs

[Ex. 5] Let $D = \{n \in \mathbb{N} \mid n > 0\} \cup \{\infty\}$ and $\sqsubseteq \subseteq (D \times D)$ such that

- for any $n, m \in D \cap \mathbb{N}$, we let $n \sqsubseteq m$ iff n divides m;
- for any $x \in D$, we let $x \sqsubseteq \infty$.

Is $(D, \sqsubseteq)$ a $\mathrm{CPO}_\bot$? Explain.

Ex. 5, divides



∞
...
16
24
8
12
18
27
30
20
28
4
6
9
15
10
25
21
14
22
26
2
3
5
7
11
13
17
19
23
29
1

Ex. 5, divides

$\text{CPO}_\perp$?

reflexive?

$d \in D$

- $d \in \mathbb{N}$ $\quad d \text{ divides } d \quad d \sqsubseteq d$
- $d = \infty$ $\quad \infty \sqsubseteq \infty$

antisymmetric?

$d_1, d_2 \in D$ $\quad d_1 \sqsubseteq d_2$, $d_2 \sqsubseteq d_1$ $\quad d_1 \stackrel{?}{=} d_2$

- $d_1, d_2 \in \mathbb{N}$ $\quad d_1 \text{ divides } d_2$, $d_2 \text{ divides } d_1$ $\quad d_1 \leq d_2$, $d_2 \leq d_1$ $\quad d_1 = d_2$
- $d_1 = \infty$ $\quad \infty \sqsubseteq d_2$ $\quad d_2 = \infty$ $\quad d_2 = \infty = d_1$
- $d_2 = \infty$ $\quad \infty \sqsubseteq d_1$ $\quad d_1 = \infty$ $\quad d_1 = \infty = d_2$

Ex. 5, divides

$$\mathrm{CPO}_\perp?$$

transitive?

$d_1, d_2, d_3 \in D$ $d_1 \sqsubseteq d_2$, $d_2 \sqsubseteq d_3$ $d_1 \overset{?}{\sqsubseteq} d_3$

- $d_1, d_2, d_3 \in \mathbb{N}$ $d_1 \text{ divides } d_2$, $d_2 \text{ divides } d_3$ $d_1 \text{ divides } d_3$ $d_1 \sqsubseteq d_3$
- $d_1 = \infty$ $d_2 = d_3 = \infty$ $\infty \sqsubseteq \infty$
- $d_2 = \infty$ $d_3 = \infty$ $d_1 \sqsubseteq \infty$
- $d_3 = \infty$ $d_1 \sqsubseteq \infty$

bottom? 1 divides every number and $1 \sqsubseteq \infty$

Ex. 5, divides

$\mathrm{CPO}_{\perp}$?

complete?

- every finite chain has a limit
- infinite chains can only contain increasing natural numbers, then the limit is ∞

Fixpoints

[Ex. 6] Define two functions $f_i : D_i \to D_i$ over two suitable CPOs D_i for $i \in [1, 2]$ (not necessarily with bottom) such that

1. f_1 is continuous, has fixpoints but not a least fixpoint;

2. f_2 is continuous and has no fixpoint;

Ex. 6, fixpoints

1. f_1 is continuous, has fixpoints but not a least fixpoint;

Let us try to find a minimal example
How many elements do we need at least?
How should they be ordered?

Ex. 6, fixpoints

1. f_1 is continuous, has fixpoints but not a least fixpoint;

$D_1 = (\{0,1\}, =)$ $\qquad\qquad$ $f_1 : D_1 \to D_1$

Discrete order: CPO

Discrete order: any function is monotone and continuous

$$f_1(0) = 0$$

$$f_1(1) = 1$$

Kleene's theorem is not applicable: why?

Ex. 6, fixpoints

2. f_2 is continuous and has no fixpoint;

Let us try to find a minimal example
How many elements do we need at least?
How should they be ordered?

Ex. 6, fixpoints

2. f_2 is continuous and has no fixpoint;

$$D_2 = D_1 = (\{0, 1\}, =) \qquad f_2 : D_2 \to D_2$$

Discrete order: CPO

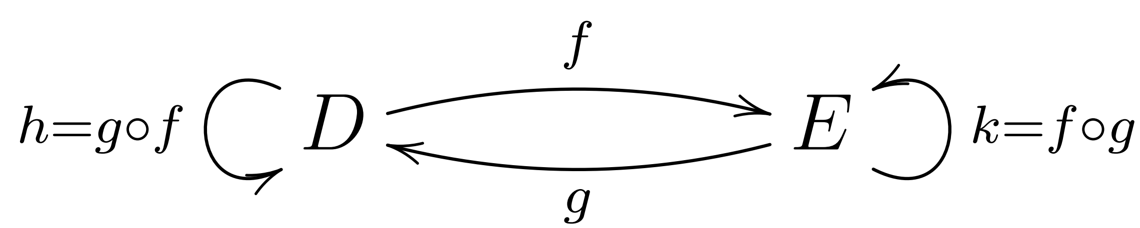
Discrete order: any function is monotone and continuous

$$f_2(0) = 1$$

$$f_2(1) = 0$$

Kleene’s theorem is not applicable: why?

[Ex. 7] Let D, E be two $\text{CPO}_\perp$s and $f : D \to E$, $g : E \to D$ be two continuous functions between them. Their compositions $h = g \circ f : D \to D$ and $k = f \circ g : E \to E$ are known to be continuous and thus have least fixpoints.

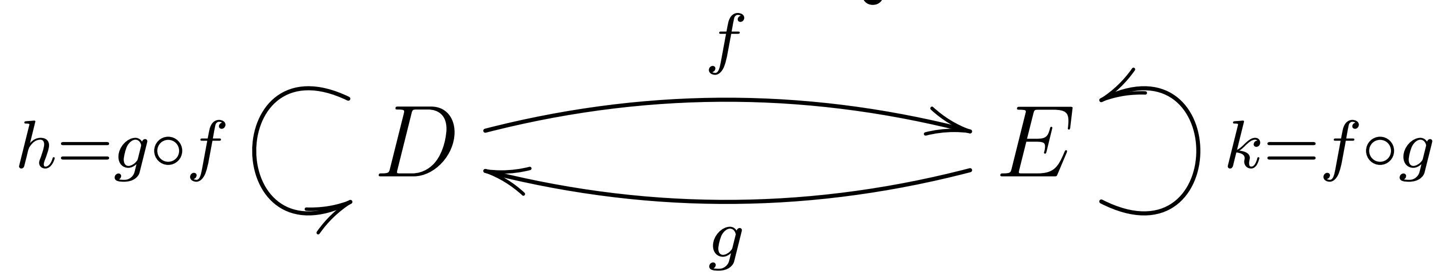


Let $e_0 = \text{fix}(k) \in E$. Prove that $g(e_0) = \text{fix}(h) \in D$ by showing that

1. $g(e_0)$ is a fixpoint for h, and

2. $g(e_0)$ is the least pre-fixpoint for h.

Ex. 7, composition



$e_0 = \mathrm{fix}(k)$

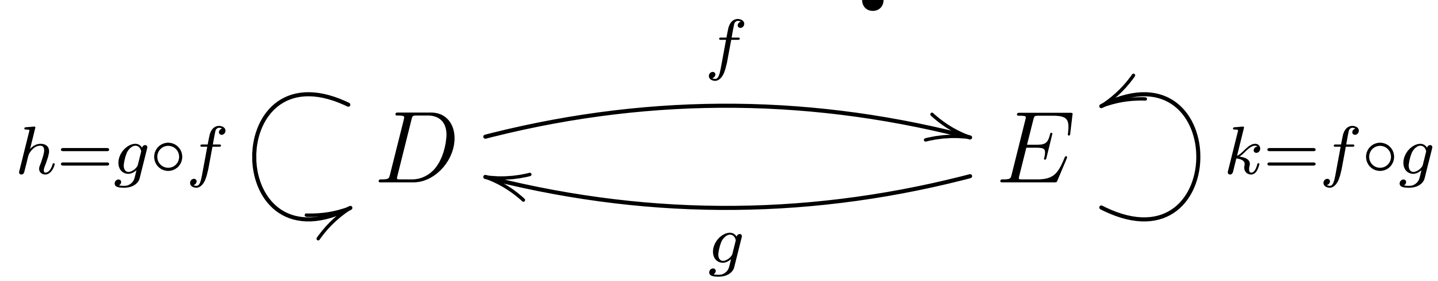
1. $g(e_0)$ is a fixpoint for h.

we need to prove $h(g(e_0)) = g(e_0)$

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

by def   by def $e_0 = \mathrm{fix}(k)$

Ex. 7, composition



$e_0 = \mathrm{fix}(k)$

2. $g(e_0)$ is the least pre-fixpoint for h.

take $d \sqsupseteq_D h(d)$ we want to prove $g(e_0) \sqsubseteq_D d$

$d \sqsupseteq_D h(d) = g(f(d))$ (by def of h)

$f(d) \sqsupseteq_E f(g(f(d))) = k(f(d))$ (by monotonicity f , def. k)

hence $f(d)$ is a pre-fixpoint of k
e_0 is the least pre-fixpoint of k
$\Rightarrow e_0 \sqsubseteq_E f(d)$

$g(e_0) \sqsubseteq_D g(f(d)) = h(d) \sqsubseteq_D d$ (by mon. g , def. h , hyp. d)