https://didawiki.di.unipi.it/doku.php/magistraleinformatica/mpp/start

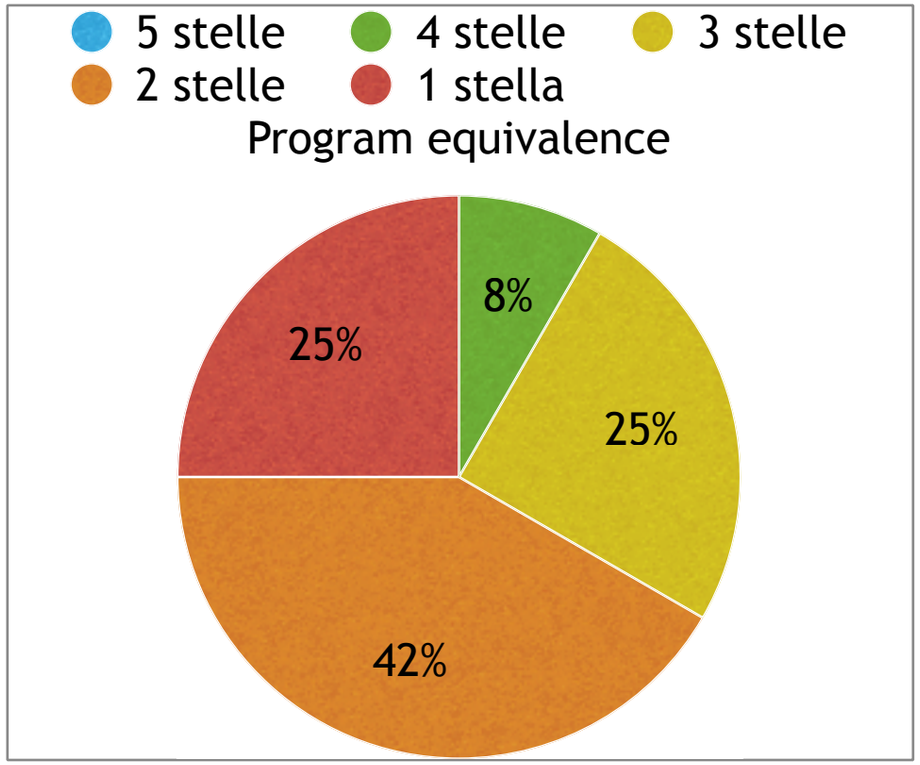
MPP 2026/27 (0077A, 9CFU)

Models for Programming Paradigms

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

06 - Operational equivalence

From your forms



(over 12 answers)

Operational equivalence

Program equivalence

Symbolic manipulation of programs is possible
if we can guarantee that their semantics is preserved

Useful to study program transformations
(interpreters, compilers, code optimisation, refactoring…)

but… when are two commands equivalent?

Operational equivalence

A first attempt

default initial memory

$\forall x.\ \sigma_0(x) = 0$

$$c_1 \sim_{\mathrm{o}} c_2 \quad \triangleq \quad \forall \sigma.\ (\ \langle c_1, \sigma_0 \rangle \longrightarrow \sigma \Leftrightarrow \langle c_2, \sigma_0 \rangle \longrightarrow \sigma\ )$$

What's wrong with this definition?

$$C[\bullet] \triangleq y := 1; [\bullet]$$

$$x := y + 1 \sim_{\mathrm{o}} x := 1$$

$$C[x := y + 1] \triangleq y := 1\,; x := y + 1$$

$$C[x := 1] \triangleq y := 1\,; x := 1$$

$$C[x := y + 1] \not\sim_{\mathrm{o}} C[x := 1]$$

Operational equivalence

any initial memory

$$c_1 \sim_{\mathrm{o}} c_2 \quad \triangleq \quad \forall \sigma, \sigma'. \; (\; \langle c_1, \sigma \rangle \longrightarrow \sigma' \Leftrightarrow \langle c_2, \sigma \rangle \longrightarrow \sigma' \;)$$

much better:
it is an equivalence relation
and also a congruence
(but we will see this later)

Concrete equivalence

$$c \overset{?}{\sim}_{\mathrm{o}} w$$

$$w \triangleq \textbf{while } x > 0 \textbf{ do } x := x - 1$$

$$c \triangleq \textbf{if } x > 0 \textbf{ then } x := 0 \textbf{ else skip}$$

Take a generic σ

if $\sigma(x) \leq 0$:

$\langle c, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow \langle x > 0, \sigma \rangle \longrightarrow \mathbf{ff}, \langle \mathbf{skip}, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow^* \langle \mathbf{skip}, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow_{\sigma' = \sigma} \square$

$\langle c, \sigma \rangle \longrightarrow \sigma$

$\langle w, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow_{\sigma' = \sigma} \langle x > 0, \sigma \rangle \longrightarrow \mathbf{ff}$

$\nwarrow^* \square$

$\langle w, \sigma \rangle \longrightarrow \sigma$

Example (ctd)

$$c \overset{?}{\sim}_{\mathrm{o}} w$$

$$w \triangleq \textbf{while } x > 0 \textbf{ do } x := x - 1$$

$$c \triangleq \textbf{if } x > 0 \textbf{ then } x := 0 \textbf{ else skip}$$

Take a generic σ if $\sigma(x) > 0$:

$\langle c, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow \langle x > 0, \sigma \rangle \longrightarrow \mathbf{tt}, \langle x := 0, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow^{*} \langle x := 0, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow^{*}_{\sigma' = \sigma[0/x]} \square$

$$\langle c, \sigma \rangle \longrightarrow \sigma[0/x]$$

$\forall n > 0.\ P(n)$ $\quad P(n) \triangleq \forall \sigma.\ \sigma(x) = n \Rightarrow \langle w, \sigma \rangle \longrightarrow \sigma[0/x]$

Example (ctd)

$$c \overset{?}{\sim}_{\mathrm{o}} w$$

$$w \triangleq \textbf{while } x > 0 \textbf{ do } x := x - 1$$

$$c \triangleq \textbf{if } x > 0 \textbf{ then } x := 0 \textbf{ else skip}$$

$$\forall n > 0.\ P(n) \qquad P(n) \triangleq \forall \sigma.\ \sigma(x) = n \Rightarrow \langle w, \sigma \rangle \longrightarrow \sigma[0/x]$$

$$P(1) \triangleq \forall \sigma.\ \sigma(x) = 1 \Rightarrow \langle w, \sigma \rangle \longrightarrow \sigma[0/x]$$

take a generic σ and suppose $\sigma(x) = 1$

$$\langle w, \sigma \rangle \longrightarrow \sigma'$$

$$\nwarrow \langle x > 0, \sigma \rangle \longrightarrow \mathbf{tt}, \langle x := x - 1, \sigma \rangle \longrightarrow \sigma'', \langle w, \sigma'' \rangle \longrightarrow \sigma'$$

$$\nwarrow^{*} \langle x := x - 1, \sigma \rangle \longrightarrow \sigma'', \langle w, \sigma'' \rangle \longrightarrow \sigma'$$

$$\nwarrow^{*}_{\sigma''=\sigma[0/x]} \langle w, \sigma[0/x] \rangle \longrightarrow \sigma'$$

$$\nwarrow_{\sigma'=\sigma[0/x]} \langle x > 0, \sigma[0/x] \rangle \longrightarrow \mathbf{ff}$$

$$\nwarrow^{*} \square \qquad \langle w, \sigma \rangle \longrightarrow \sigma[0/x]$$

Example (ctd)

$$c \overset{?}{\sim}_{\mathrm{o}} w$$

$$w \triangleq \textbf{while } x > 0 \textbf{ do } x := x - 1$$

$$c \triangleq \textbf{if } x > 0 \textbf{ then } x := 0 \textbf{ else skip}$$

$\forall n > 0.\ P(n)$ $\qquad P(n) \triangleq \forall \sigma.\ \sigma(x) = n \Rightarrow \langle w, \sigma \rangle \longrightarrow \sigma[0/x]$

$\forall n > 0.\ P(n) \Rightarrow P(n+1)$ take a generic $n > 0$

we assume $P(n) \triangleq \forall \sigma.\ \sigma(x) = n \Rightarrow \langle w, \sigma \rangle \longrightarrow \sigma[0/x]$

we prove $P(n+1) \triangleq \forall \sigma.\ \sigma(x) = n+1 \Rightarrow \langle w, \sigma \rangle \longrightarrow \sigma[0/x]$

take a generic σ and suppose $\sigma(x) = n+1$

$\langle w, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow \langle x > 0, \sigma \rangle \longrightarrow \mathbf{tt}, \langle x := x - 1, \sigma \rangle \longrightarrow \sigma'', \langle w, \sigma'' \rangle \longrightarrow \sigma'$

$\nwarrow^{*} \langle x := x - 1, \sigma \rangle \longrightarrow \sigma'', \langle w, \sigma'' \rangle \longrightarrow \sigma'$

$\nwarrow^{*}_{\sigma''=\sigma[n/x]} \langle w, \sigma[n/x] \rangle \longrightarrow \sigma'$

by inductive hypothesis $P(n)$ we know $\langle w, \sigma[n/x] \rangle \longrightarrow \sigma[0/x]$

by determinacy it must be $\sigma' = \sigma[0/x]$ $\qquad \langle w, \sigma \rangle \longrightarrow \sigma[0/x]$

Parametric equivalence

$$w \overset{?}{\sim}_{\mathrm{o}} ww$$

$$w \triangleq \textbf{while } b \textbf{ do } c$$

$$ww \triangleq \textbf{while } b \textbf{ do } (\overbrace{\textbf{while } b \textbf{ do } c}^{w})$$

Take a generic σ if $\langle b, \sigma \rangle \longrightarrow \mathbf{ff}$:

$\langle w, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow_{\sigma' = \sigma} \langle b, \sigma \rangle \longrightarrow \mathbf{ff}$

$\nwarrow^{*} \square$ $\langle w, \sigma \rangle \longrightarrow \sigma$

$\langle ww, \sigma \rangle \longrightarrow \sigma'$

$\nwarrow_{\sigma' = \sigma} \langle b, \sigma \rangle \longrightarrow \mathbf{ff}$

$\nwarrow^{*} \square$ $\langle ww, \sigma \rangle \longrightarrow \sigma$

Example (ctd)

$$w \overset{?}{\sim}_{\mathrm{o}} ww$$

$$w \triangleq \textbf{while } b \textbf{ do } c$$

$$ww \triangleq \textbf{while } b \textbf{ do } (\overbrace{\textbf{while } b \textbf{ do } c}^{w})$$

Take a generic σ if $\langle b, \sigma \rangle \longrightarrow \mathbf{tt}$: if $\langle w, \sigma \rangle \longrightarrow \sigma'$:

$$\langle ww, \sigma \rangle \longrightarrow \sigma'_1$$

$$\nwarrow \langle b, \sigma \rangle \longrightarrow \mathbf{tt}, \langle w, \sigma \rangle \longrightarrow \sigma'', \langle ww, \sigma'' \rangle \longrightarrow \sigma'_1$$

$$\nwarrow^* \langle w, \sigma \rangle \longrightarrow \sigma'', \langle ww, \sigma'' \rangle \longrightarrow \sigma'_1$$

$$\nwarrow^*_{\sigma''=\sigma'} \langle ww, \sigma' \rangle \longrightarrow \sigma'_1$$

$$\nwarrow_{\sigma'_1=\sigma'} \langle b, \sigma' \rangle \longrightarrow \mathbf{ff}$$

$$\nwarrow^* \square$$

$$\langle ww, \sigma \rangle \longrightarrow \sigma'$$

see previous exercise

$$\forall b, c, \sigma, \sigma'. \ \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \longrightarrow \sigma' \Rightarrow \langle b, \sigma' \rangle \longrightarrow \mathbf{ff}$$

Example (ctd)

$$w \overset{?}{\sim}_{\mathrm{o}} ww$$

$$w \triangleq \textbf{while } b \textbf{ do } c$$

$$ww \triangleq \textbf{while } b \textbf{ do } (\overbrace{\textbf{while } b \textbf{ do } c}^{w})$$

Take a generic σ if $\langle b, \sigma \rangle \longrightarrow \mathbf{tt}$: if $\langle w, \sigma \rangle \not\longrightarrow$:

$\langle ww, \sigma \rangle \longrightarrow \sigma'_1$

$\nwarrow \langle b, \sigma \rangle \longrightarrow \mathbf{tt}, \langle w, \sigma \rangle \longrightarrow \sigma'', \langle ww, \sigma'' \rangle \longrightarrow \sigma'_1$

$\nwarrow^* \langle w, \sigma \rangle \longrightarrow \sigma'', \langle ww, \sigma'' \rangle \longrightarrow \sigma'_1$

will diverge, because by hypothesis $\langle w, \sigma \rangle \not\longrightarrow$

$\langle ww, \sigma \rangle \not\longrightarrow$

A note on divergence

any two (always) divergent commands are equivalent

$w_1 \overset{?}{\sim}_{\mathrm{o}} w_2$ ✓ $\quad w_1 \overset{?}{\sim}_{\mathrm{o}} w_3$ ✓ $\quad w_1 \overset{?}{\sim}_{\mathrm{o}} w_4$ ✓

$$w_1 \triangleq \textbf{while true do skip}$$

$$w_2 \triangleq x := 0\,;\textbf{while } x \geq 0 \textbf{ do } x := x + 1$$

$$w_3 \triangleq y := 0\,;\textbf{while } y \leq 0 \textbf{ do } y := y - 100$$

$$w_4 \triangleq y := x + 1\,;\textbf{while } x \neq y \textbf{ do } (x := x - 1\,; y := y + 1)$$

A difficult problem

$w_1 \triangleq$ **while true do skip**

$$
\begin{array}{ll}
p \triangleq & x := 3; \\
& s := 1; \\
& \textbf{while } x \neq s \textbf{ do } (\\
& \qquad x := x + 2; \\
& \qquad s := 1; \\
& \qquad i := 2; \\
& \qquad \textbf{while } 2 \times i \leq x \textbf{ do } (\\
& \qquad\qquad \textbf{if } x \% i = 0 \textbf{ then } s := s + i \\
& \qquad\qquad\qquad\qquad \textbf{else skip}; \\
& \qquad\qquad i := i + 1 \\
& \qquad) \\
&)
\end{array}
$$

$$w_1 \overset{?}{\sim}_{\mathrm{o}} p$$

A difficult problem

$w_1 \triangleq$ **while true do skip**

$w_1 \overset{?}{\sim}_{\mathrm{o}} p$

always terminate

when $x \geq 2$, at the end, s takes the sum of all proper divisors of x

$$
\begin{array}{l}
s := 1; \\
i := 2; \\
\textbf{while } 2 \times i \leq x \textbf{ do } (\\
\qquad \textbf{if } x\%i = 0 \textbf{ then } s := s + i \\
\qquad\qquad\qquad\qquad \textbf{else skip}; \\
\qquad i := i + 1 \\
)
\end{array}
$$

A difficult problem

$w_1 \triangleq$ **while true do skip**

$w_1 \overset{?}{\sim}_{\mathrm{o}} p$

$$s := \sum_{i \in \mathsf{div}(x)} i$$

where $\mathsf{div}(x) \triangleq \{1\} \cup \{d \mid 1 < d < x, x\%d = 0\}$

$$i := 1 + (x/2)$$

A difficult problem

$w_1 \triangleq$ **while true do skip**

$p \triangleq$ $x := 3;$ an odd number

$s := 1;$ sum of proper divisors of x

while $x \neq s$ **do** (exit when x is perfect

$x := x + 2;$ next odd number

$$s := \sum_{i \in \mathsf{div}(x)} i$$ sum of proper divisors of x

where $\mathsf{div}(x) \triangleq \{1\} \cup \{d \mid 1 < d < x, x\%d = 0\}$

$$i := 1 + (x/2)$$

)

$w_1 \overset{?}{\sim}_{\mathrm{o}} p$

a number is perfect

if it equals the sum of its proper divisors

$6 = 1 + 2 + 3$

A difficult problem

$w_1 \triangleq$ **while true do skip**

$p \triangleq$ terminate when the first odd perfect number is found

$w_1 \overset{?}{\sim}_{\mathrm{o}} p$

as of 2022
it is not known if any odd perfect number exists

Ochem, Pascal; Rao, Michaël.
Odd perfect numbers are greater than 10^{1500}.
Mathematics of Computation n.81(279): 1869–1877.
(2012) doi:10.1090/S0025-5718-2012-02563-4

a number is perfect
if it equals the sum of its proper divisors

For the curious minds

Euclid proved that $2^{p-1}(2^p - 1)$ is an even perfect number whenever $2^p - 1$ is prime

Euler proved that any even perfect number has the form $2^{p-1}(2^p - 1)$ with $2^p - 1$ prime

Prime numbers of the form $2^p - 1$ are now called Mersenne primes

Even perfect numbers and Mersenne primes are thus in 1-to-1 correspondence

For $2^p - 1$ to be prime it is necessary that p is prime

If p is prime, $2^p - 1$ is not necessarily prime

update: in 2024 a 52th Mersenne prime was discovered!

As of 2022, there are only 51 Mersenne primes known

It is not known if there are infinitely many Mersenne primes

For the curious minds

prime number p	Mersenne prime $2^p - 1$	even perfect number $2^{p-1}(2^p - 1)$
2	3	6
3	7	28
5	31	496
7	127	8.128
13	8.191	33.550.336
17	131.071	8.589.869.056
...	...	...
82.589.933	24.862.048 digits!	49.724.095 digits!
136.279.841	41.024.320 digits!	82.048.640 digits!

in binary form:
p ones followed
by $p - 1$ zeros

GIMPS on October 12, 2024

A tricky problem

$w_1 \triangleq$ **while true do skip**

$$
\begin{array}{ll}
p_1 \triangleq & x := 2; \\
& s := 1; \\
& \textbf{while } x \neq s \textbf{ do } (\\
& \qquad x := x + 2; \\
& \qquad s := 1; \\
& \qquad i := 2; \\
& \qquad \textbf{while } 2 \times i \leq x \textbf{ do } (\\
& \qquad\qquad \textbf{if } x\%i = 0 \textbf{ then } s := s + i \\
& \qquad\qquad\qquad\qquad \textbf{else skip}; \\
& \qquad\qquad i := i + 1 \\
& \qquad) \\
&)
\end{array}
$$

$w_1 \overset{?}{\sim}_{\mathrm{o}} p_1$ ✗

$x := 6 \overset{?}{\sim}_{\mathrm{o}} p_1$ ✗

$$
\begin{array}{l}
x := 6; \\
s := 6; \\
i := 4;
\end{array}
\overset{?}{\sim}_{\mathrm{o}} p_1
$$ ✓

A variant

$w_1 \triangleq$ **while true do skip**

$$w_1 \overset{?}{\sim}_{\mathrm{o}} p_2$$

$$\begin{array}{ll}
p_2 \triangleq & s := 1; \\
& \textbf{while } x \neq s \textbf{ do } (\\
& \qquad x := x + 2; \\
& \qquad s := 1; \\
& \qquad i := 2; \\
& \qquad \textbf{while } 2 \times i \leq x \textbf{ do } (\\
& \qquad\qquad \textbf{if } x \% i = 0 \textbf{ then } s := s + i \\
& \qquad\qquad\qquad\qquad\qquad \textbf{else skip}; \\
& \qquad\qquad i := i + 1 \\
& \qquad) \\
&)
\end{array}$$

$$\begin{array}{ll}
x := 6; & \\
s := 6; & \overset{?}{\sim}_{\mathrm{o}} p_2 \\
i := 4; &
\end{array}$$

A last instance

$$p_2 \overset{?}{\sim}_{\mathrm{o}} p_3$$

$$\times$$

$$
\begin{array}{ll}
p_2 \triangleq & s := 1; \\
& \textbf{while } x \neq s \textbf{ do } (\\
& \quad x := x + 2; \\
& \quad s := 1; \\
& \quad i := 2; \\
& \quad \textbf{while } 2 \times i \leq x \textbf{ do } (\\
& \qquad \textbf{if } x\%i = 0 \textbf{ then } s := s + i \\
& \qquad\qquad\qquad \textbf{else skip}; \\
& \qquad i := i + 1 \\
& \quad) \\
&) \\
\end{array}
$$

$$
\begin{array}{ll}
p_3 \triangleq & s := 1; \\
& \textbf{while } x \neq s \textbf{ do } (\\
& \quad x := x + 1; \\
& \quad s := 1; \\
& \quad i := 2; \\
& \quad \textbf{while } 2 \times i \leq x \textbf{ do } (\\
& \qquad \textbf{if } x\%i = 0 \textbf{ then } s := s + i \\
& \qquad\qquad\qquad \textbf{else skip}; \\
& \qquad i := i + 1 \\
& \quad) \\
&) \\
\end{array}
$$