Old Program Theory

Hoare Logic, 1969

h:= 0. j:= #L.

while j–h > 1

do

i:= (h+j)/2.

if L i ≤ x **then** h:= i **else** j:= i **fi**

od

Old Program Theory

Hoare Logic, 1969

$\{\ \top\ \}$

$h := 0.\ \ j := \#L.$

$\{\ h<j \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x)\ \}$

while $j-h > 1$

$\{\ j-h > 1 \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x)\ \}$

do $\{\ 0 < j-h = V\ \}$

$i := (h+j)/2.$

if $L\ i \le x$ **then** $h := i$ **else** $j := i$ **fi**

$\{\ h<j \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x)\ \}$

$\{\ 0 \le j-h < V\ \}$ **od**

$\{\ j-h \le 1 \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x)\ \}$

$\{\ (\exists i\colon 0,..\#L\cdot L\ i = x)\ \ =\ \ (L\ h = x)\ \}$

Old Program Theory

Hoare Logic, 1969

$\{ \top \}$ ⟵ precondition

$h := 0.\ \ j := \#L.$

$\{ h<j \wedge \neg(\exists i\colon 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot\ L\ i = x) \}$

while $j{-}h > 1$

$\{ j{-}h > 1 \wedge \neg(\exists i\colon 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot\ L\ i = x) \}$

do $\{ 0 < j{-}h = V \}$

$i := (h{+}j)/2.$

if $L\ i \le x$ **then** $h := i$ **else** $j := i$ **fi**

$\{ h<j \wedge \neg(\exists i\colon 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot\ L\ i = x) \}$

$\{ 0 \le j{-}h < V \}$ **od**

$\{ j{-}h \le 1 \wedge \neg(\exists i\colon 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot\ L\ i = x) \}$

$\{ (\exists i\colon 0,..\#L\cdot\ L\ i = x) \ = \ (L\ h = x) \}$

Old Program Theory

Hoare Logic, 1969

$\{ \top \}$ ⟵ precondition

$h{:=}\ 0.\ \ j{:=}\ \#L.$

$\{\ h<j \wedge \neg(\exists i{:}\ 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i{:}\ j,..\#L\cdot\ L\ i = x)\ \}$

while $j–h > 1$

$\{\ j–h > 1 \wedge \neg(\exists i{:}\ 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i{:}\ j,..\#L\cdot\ L\ i = x)\ \}$

do $\{\ 0 < j–h = V\ \}$

$i{:=}\ (h+j)/2.$

if $L\ i \le x$ **then** $h{:=}\ i$ **else** $j{:=}\ i$ **fi**

$\{\ h<j \wedge \neg(\exists i{:}\ 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i{:}\ j,..\#L\cdot\ L\ i = x)\ \}$

$\{\ 0 \le j–h < V\ \}$ **od**

$\{\ j–h \le 1 \wedge \neg(\exists i{:}\ 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i{:}\ j,..\#L\cdot\ L\ i = x)\ \}$

$\{\ (\exists i{:}\ 0,..\#L\cdot\ L\ i = x)\ \ =\ \ (L\ h = x)\ \}$ ⟵ postcondition

Old Program Theory

Hoare Logic, 1969

$\{ \top \}$ ← precondition

$h{:=}\,0.\ \ j{:=}\,\#L.$

$\{\ h<j \wedge \neg(\exists i{:}\ 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i{:}\ j,..\#L\cdot\ L\ i = x)\ \}$ ← invariant

while $j{-}h > 1$

$\{\ j{-}h > 1 \wedge \neg(\exists i{:}\ 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i{:}\ j,..\#L\cdot\ L\ i = x)\ \}$

do $\{\ 0 < j{-}h = V\ \}$

$i{:=}\,(h{+}j)/2.$

if $L\ i \le x$ **then** $h{:=}\,i$ **else** $j{:=}\,i$ **fi**

$\{\ h<j \wedge \neg(\exists i{:}\ 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i{:}\ j,..\#L\cdot\ L\ i = x)\ \}$ ← invariant

$\{\ 0 \le j{-}h < V\ \}$ **od**

$\{\ j{-}h \le 1 \wedge \neg(\exists i{:}\ 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i{:}\ j,..\#L\cdot\ L\ i = x)\ \}$

$\{\ (\exists i{:}\ 0,..\#L\cdot\ L\ i = x)\ \ =\ \ (L\ h = x)\ \}$ ← postcondition

Old Program Theory

Hoare Logic, 1969

$\{ \top \}$ ← precondition

h:= 0. j:= #L.

$\{ h<j \wedge \neg(\exists i\text{: } 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i\text{: } j,..\#L\cdot\ L\ i = x) \}$ ← invariant

while $j\text{–}h > 1$

$\{ j\text{–}h > 1 \wedge \neg(\exists i\text{: } 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i\text{: } j,..\#L\cdot\ L\ i = x) \}$

do $\{ 0 < j\text{–}h = V \}$ ← variant

i:= $(h+j)/2$.

if $L\ i \le x$ **then** h:= i **else** j:= i **fi**

$\{ h<j \wedge \neg(\exists i\text{: } 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i\text{: } j,..\#L\cdot\ L\ i = x) \}$ ← invariant

$\{ 0 \le j\text{–}h < V \}$ **od**

$\{ j\text{–}h \le 1 \wedge \neg(\exists i\text{: } 0,..h\cdot\ L\ i = x) \wedge \neg(\exists i\text{: } j,..\#L\cdot\ L\ i = x) \}$

$\{ (\exists i\text{: } 0,..\#L\cdot\ L\ i = x) \ =\ (L\ h = x) \}$ ← postcondition

Old Program Theory

Hoare Logic, 1969

$\{ \top \}$ ⟵ precondition

$h:= 0.\ \ j:= \#L.$

$\{ h<j \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x) \}$ ⟵ invariant

while $j–h > 1$

$\{ j–h > 1 \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x) \}$

do $\{ 0 < j–h = V \}$ ⟵ variant

$i:= (h+j)/2.$

if $L\ i \le x$ **then** $h:= i$ **else** $j:= i$ **fi**

$\{ h<j \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x) \}$ ⟵ invariant

$\{ 0 \le j–h < V \}$ ⟵ variant **od**

$\{ j–h \le 1 \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x) \}$

$\{ (\exists i\colon 0,..\#L\cdot L\ i = x) \ = \ (L\ h = x) \}$ ⟵ postcondition

Old Program Theory

Hoare Logic, 1969

$\{ \top \}$ ← precondition

$h := 0.\ \ j := \#L.$

$\{ h<j \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x) \}$ ← invariant

while $j–h > 1$

$\{ j–h > 1 \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x) \}$ ← assertion

do $\{ 0 < j–h = V \}$ ← variant

$i := (h+j)/2.$

if $L\ i \le x$ **then** $h := i$ **else** $j := i$ **fi**

$\{ h<j \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x) \}$ ← invariant

$\{ 0 \le j–h < V \}$ ← variant **od**

$\{ j–h \le 1 \wedge \neg(\exists i\colon 0,..h\cdot L\ i = x) \wedge \neg(\exists i\colon j,..\#L\cdot L\ i = x) \}$ ← assertion

$\{ (\exists i\colon 0,..\#L\cdot L\ i = x)\ \ =\ \ (L\ h = x) \}$ ← postcondition

exact precondition for P to be refined by S : $\forall\sigma'\cdot P\Leftarrow S$

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

(the exact precondition for $x'>5$ to be refined by $x:=x+1$)

exact precondition for P to be refined by S : $\forall\sigma'\cdot P\Leftarrow S$

(the exact precondition for $x'>5$ to be refined by $x:=x+1$)

$=$ $\forall x'\cdot x'>5 \Leftarrow (x:=x+1)$

exact precondition for P to be refined by S : $\forall\sigma'\cdot P\Leftarrow S$

(the exact precondition for $x'>5$ to be refined by $x:=x+1$)

$=\quad \forall x'\cdot x'>5 \Leftarrow (x:=x+1)$ expand assignment

$=\quad \forall x'\cdot x'>5 \Leftarrow x'=x+1$

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

(the exact precondition for $x'>5$ to be refined by $x:= x+1$)

$=\quad \forall x'\cdot x'>5 \Leftarrow (x:= x+1)$ expand assignment

$=\quad \forall x'\cdot x'>5 \Leftarrow x'=x+1$ One-Point Law

$=\quad x+1 > 5$

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

(the exact precondition for $x'>5$ to be refined by $x:= x+1$)

$=$	$\forall x'\cdot x'>5 \Leftarrow (x:= x+1)$	expand assignment
$=$	$\forall x'\cdot x'>5 \Leftarrow x'=x+1$	One-Point Law
$=$	$x+1 > 5$	simplify
$=$	$x > 4$	

exact precondition for P to be refined by S : $\forall\sigma' \cdot P \Leftarrow S$

(the exact precondition for $x'>5$ to be refined by $x:= x+1$)

$=$	$\forall x' \cdot x'>5 \Leftarrow (x:= x+1)$	expand assignment
$=$	$\forall x' \cdot x'>5 \Leftarrow x'=x+1$	One-Point Law
$=$	$x+1 > 5$	simplify
$=$	$x > 4$	

$x'>5 \quad \Leftarrow \quad x:= x+1$ ✗

exact precondition for P to be refined by S : $\forall \sigma' \cdot P \Leftarrow S$

(the exact precondition for $x'{>}5$ to be refined by $x{:=}\,x{+}1$)

$=$ $\forall x' \cdot x'{>}5 \Leftarrow (x{:=}\,x{+}1)$ expand assignment

$=$ $\forall x' \cdot x'{>}5 \Leftarrow x'{=}x{+}1$ One-Point Law

$=$ $x{+}1 > 5$ simplify

$=$ $x > 4$

$x'{>}5 \;\Longleftarrow\; x{:=}\,x{+}1$ ✗

$x{>}4 \Rightarrow x'{>}5 \;\Longleftarrow\; x{:=}\,x{+}1$ ✔

exact precondition for P to be refined by S : $\forall\sigma'\cdot P\Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P\Leftarrow S$

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P \Leftarrow S$

(the exact postcondition for $x>4$ to be refined by $x:=x+1$)

exact precondition for P to be refined by S : $\forall \sigma' \cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall \sigma \cdot P \Leftarrow S$

(the exact postcondition for $x>4$ to be refined by $x := x+1$)

$=$ $\forall x \cdot x>4 \Leftarrow (x := x+1)$

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P \Leftarrow S$

(the exact postcondition for $x>4$ to be refined by $x:= x+1$)

$=$ $\forall x\cdot x>4 \Leftarrow (x:= x+1)$ expand assignment

$=$ $\forall x\cdot x>4 \Leftarrow x'=x+1$

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P \Leftarrow S$

(the exact postcondition for $x>4$ to be refined by $x:= x+1$)

=	$\forall x\cdot\ x>4 \Leftarrow (x:= x+1)$	expand assignment
=	$\forall x\cdot\ x>4 \Leftarrow x'=x+1$	
=	$\forall x\cdot\ x>4 \Leftarrow x=x'-1$	

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P \Leftarrow S$

(the exact postcondition for $x>4$ to be refined by $x{:=}\ x+1$)

$=\quad \forall x\cdot\ x>4 \Leftarrow (x{:=}\ x+1)$ expand assignment

$=\quad \forall x\cdot\ x>4 \Leftarrow x'=x+1$

$=\quad \forall x\cdot\ x>4 \Leftarrow x=x'-1$ One-Point Law

$=\quad x'-1 > 4$

exact precondition for P to be refined by S : $\forall\sigma' \cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma \cdot P \Leftarrow S$

	(the exact postcondition for $x>4$ to be refined by $x:= x+1$)	
=	$\forall x \cdot x>4 \Leftarrow (x:= x+1)$	expand assignment
=	$\forall x \cdot x>4 \Leftarrow x'=x+1$	
=	$\forall x \cdot x>4 \Leftarrow x=x'-1$	One-Point Law
=	$x'-1 > 4$	simplify
=	$x' > 5$	

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P \Leftarrow S$

	(the exact postcondition for $x>4$ to be refined by $x{:=}\ x+1$)	
=	$\forall x\cdot\ x>4 \Leftarrow (x{:=}\ x+1)$	expand assignment
=	$\forall x\cdot\ x>4 \Leftarrow x'=x+1$	
=	$\forall x\cdot\ x>4 \Leftarrow x=x'-1$	One-Point Law
=	$x'-1 > 4$	simplify
=	$x' > 5$	

$x>4 \quad \Leftarrow \quad x{:=}\ x+1$ ✗

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P \Leftarrow S$

(the exact postcondition for $x>4$ to be refined by $x:=x+1$)

$=$	$\forall x\cdot\ x>4 \Leftarrow (x:=x+1)$	expand assignment
$=$	$\forall x\cdot\ x>4 \Leftarrow x'=x+1$	
$=$	$\forall x\cdot\ x>4 \Leftarrow x=x'-1$	One-Point Law
$=$	$x'-1 > 4$	simplify
$=$	$x' > 5$	

$x>4 \;\;\Leftarrow\;\; x:=x+1$ ✗

$x'>5 \Rightarrow x>4 \;\;\Leftarrow\;\; x:=x+1$ ✔

exact precondition for P to be refined by S : $\forall\sigma'\cdot P \Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P \Leftarrow S$

	(the exact postcondition for $x{>}4$ to be refined by $x{:=}\ x{+}1$)	
$=$	$\forall x\cdot\ x{>}4 \Leftarrow (x{:=}\ x{+}1)$	expand assignment
$=$	$\forall x\cdot\ x{>}4 \Leftarrow x'{=}x{+}1$	
$=$	$\forall x\cdot\ x{>}4 \Leftarrow x{=}x'{-}1$	One-Point Law
$=$	$x'{-}1 > 4$	simplify
$=$	$x' > 5$	

$x{>}4 \quad \Longleftarrow \quad x{:=}\ x{+}1$ ✗

$x'{>}5 \Rightarrow x{>}4 \quad \Longleftarrow \quad x{:=}\ x{+}1$ ✔

$x{\le}4 \Rightarrow x'{\le}5 \quad \Longleftarrow \quad x{:=}\ x{+}1$ ✔

exact precondition for P to be refined by S : $\forall\sigma'\cdot P\Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P\Leftarrow S$

sufficient precondition $\Rightarrow$ exact precondition $\Rightarrow$ necessary precondition

sufficient postcondition $\Rightarrow$ exact postcondition $\Rightarrow$ necessary postcondition

exact precondition for P to be refined by S : $\forall\sigma'\cdot P\Leftarrow S$

exact postcondition for P to be refined by S : $\forall\sigma\cdot P\Leftarrow S$

sufficient precondition $\Rightarrow$ exact precondition $\Rightarrow$ necessary precondition

sufficient postcondition $\Rightarrow$ exact postcondition $\Rightarrow$ necessary postcondition

precondition law

C is a sufficient precondition for P to be refined by S

if and only if $C\Rightarrow P$ is refined by S .

postcondition law

C' is a sufficient postcondition for P to be refined by S

if and only if $C'\Rightarrow P$ is refined by S .

invariant

Let S be a specification.

Let I be an assertion with all variables unprimed.

Let I' be the same as I but with primes on all variables.

I is an invariant for S if $I \Rightarrow I'$ is refined by S .

$$\forall \sigma, \sigma' \cdot (I \Rightarrow I') \Leftarrow S$$

invariant

Let S be a specification.

Let I be an assertion with all variables unprimed.

Let I' be the same as I but with primes on all variables.

I is an invariant for S if $I \Rightarrow I'$ is refined by S .

$\forall \sigma, \sigma' \cdot (I \Rightarrow I') \Leftarrow S$

$\forall \sigma, \sigma' \cdot S \land I \Rightarrow I'$

Let the variables be x and y .

Prove that $y=x^2$ is an invariant for $(x:= x+1.\ y:= y + 2\times x - 1)$.

Let the variables be x and y .

Prove that $y=x^2$ is an invariant for $(x{:=}\ x+1.\ \ y{:=}\ y + 2\times x - 1)$.

$$(I \Rightarrow I') \Leftarrow S$$

Let the variables be x and y .

Prove that $y=x^2$ is an invariant for $(x:= x+1.\ \ y:= y + 2\times x - 1)$.

	$(I\Rightarrow I') \Leftarrow S$	replace I and S
$=$	$(y=x^2 \Rightarrow y'=x'^2) \Leftarrow (x:= x+1.\ \ y:= y + 2\times x - 1)$	

Let the variables be x and y .

Prove that $y{=}x^2$ is an invariant for $(x{:=}\ x{+}1.\ \ y{:=}\ y + 2{\times}x - 1)$.

	$(I{\Rightarrow}I') \Leftarrow S$	replace I and S
$=$	$(y{=}x^2 \Rightarrow y'{=}x'^2) \Leftarrow (x{:=}\ x{+}1.\ \ y{:=}\ y + 2{\times}x - 1)$	replace last assignment
$=$	$(y{=}x^2 \Rightarrow y'{=}x'^2) \Leftarrow (x{:=}\ x{+}1.\ \ x'{=}x\ \wedge\ y' = y + 2{\times}x - 1)$	

Let the variables be x and y .

Prove that $y=x^2$ is an invariant for $(x:= x+1.\ \ y:= y + 2\times x - 1)$.

$$(I\Rightarrow I') \Leftarrow S \qquad \text{replace } I \text{ and } S$$

$$= \quad (y=x^2 \Rightarrow y'=x'^2) \Leftarrow (x:= x+1.\ \ y:= y + 2\times x - 1) \qquad \text{replace last assignment}$$

$$= \quad (y=x^2 \Rightarrow y'=x'^2) \Leftarrow (x:= x+1.\ \ x'=x \ \wedge\ y' = y + 2\times x - 1) \qquad \text{substitution}$$

$$= \quad (y=x^2 \Rightarrow y'=x'^2) \ \Leftarrow\ x'=x+1 \ \wedge\ y' = y + 2\times(x+1) - 1$$

Let the variables be x and y .

Prove that $y=x^2$ is an invariant for $(x:= x+1.\ \ y:= y + 2\times x - 1)$.

	$(I\Rightarrow I') \Leftarrow S$	replace I and S
$=$	$(y=x^2 \Rightarrow y'=x'^2) \Leftarrow (x:= x+1.\ \ y:= y + 2\times x - 1)$	replace last assignment
$=$	$(y=x^2 \Rightarrow y'=x'^2) \Leftarrow (x:= x+1.\ \ x'=x \ \land\ y' = y + 2\times x - 1)$	substitution
$=$	$(y=x^2 \Rightarrow y'=x'^2) \ \Leftarrow\ x'=x+1 \ \land\ y' = y + 2\times(x+1) - 1$	arithmetic
$=$	$(y=x^2 \Rightarrow y'=x'^2) \ \Leftarrow\ x'=x+1 \ \land\ y' = y + 2\times x + 1$	context
$=$	$(y=x^2 \Rightarrow (y + 2\times x + 1)=(x+1)^2) \ \Leftarrow\ x'=x+1 \ \land\ y' = y + 2\times x + 1$	arithmetic
$=$	$(y=x^2 \Rightarrow (y + 2\times x + 1)=(x+1)^2) \ \Leftarrow\ x'=x+1 \ \land\ y' = y + 2\times x + 1$	
		arithmetic and cancellation
$=$	$(y=x^2 \Rightarrow y=x^2) \ \Leftarrow\ x'=x+1 \ \land\ y' = y + 2\times x + 1$	reflexive, base
$=$	$\top$	

variant

time bound, recursive measure, clock runs backwards