michelepasqua.github.io/

michele.pasqua@univr.it

UNIVERSITÀ di VERONA

Dipartimento di INFORMATICA

Principles and Applications of Abstract Interpretation

Static Analysis

Michele Pasqua, PhD

Verona, IT

October 2024

PhD Course @ UniVR 2024/2025

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

Static Analysis by Abstract Interpretation

Ingredients for a sound static analysis

Start from the denotation of program executions (e.g., sequences of states)

Ingredients for a sound static analysis

Start from the denotation of program executions (e.g., sequences of states)

Program Property

Program aspects to prove

(e.g., avoid unsafe states)

Ingredients for a sound static analysis

Start from the denotation of program executions (e.g., sequences of states)

Program aspects to prove

(e.g., avoid unsafe states)

Program Property

Concrete Semantics

Formal model of programs behavior

(e.g., sets of reachable states)

Ingredients for a sound static analysis

Start from the denotation of program executions (e.g., sequences of states)

Program aspects to prove
(e.g., avoid unsafe states)

Program Property

Concrete Semantics

Abstract Semantics

Formal model of programs behavior
(e.g., sets of reachable states)

Discard proof-irrelevant information
(e.g., numeric abstraction)

Ingredients for a sound static analysis

Start from the denotation of program executions (e.g., sequences of states)

Program aspects to prove
(e.g., avoid unsafe states)

Program Property

Terminating Static Analysis

Automated proof

Concrete Semantics

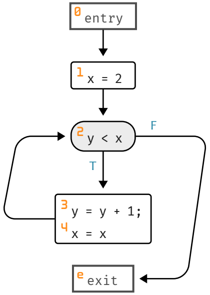
Abstract Semantics

Formal model of programs behavior
(e.g., sets of reachable states)

Discard proof-irrelevant information
(e.g., numeric abstraction)

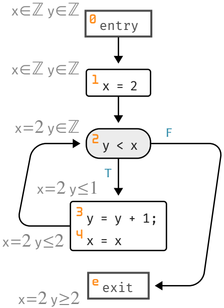
Compute an invariant on program variables at a program point

```
1 x = 2;
2 while (y < x) {
    3 y = y + 1;
    4 x = x
  }
```



Compute an invariant on program variables at a program point

```
@{ x:ℤ, y:ℤ }
1 x = 2;  @{ x:ℤ, y:ℤ }
2 while (y < x) {  @{ x:{2}, y:ℤ }
    3 y = y + 1;  @{ x:{2}, y:{z ∈ ℤ | z ≤ 1} }
    4 x = x  @{ x:{2}, y:{z ∈ ℤ | z ≤ 2} }
  }
@{ x:{2}, y:{z ∈ ℤ | z ≥ 2} }
```



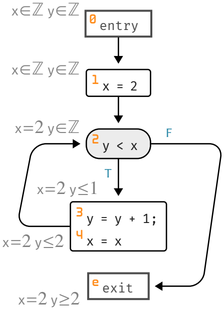
Compute an invariant on program variables at a program point

- To check the (un)reachability of unsafe states

$@\{x{:}\mathbb{Z}, y{:}\mathbb{Z}\}$

```
1 x = 2;        @{x:ℤ, y:ℤ}
2 while (y < x) {   @{x:{2}, y:ℤ}
    3 y = y + 1;    @{x:{2}, y:{z ∈ ℤ | z ≤ 1}}
    4 x = x      @{x:{2}, y:{z ∈ ℤ | z ≤ 2}}
  }
```

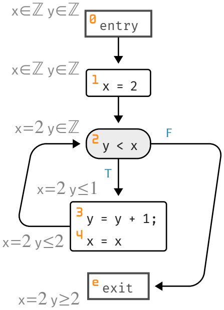
$@\{x{:}\{2\}, y{:}\{z \in \mathbb{Z} \mid z \geq 2\}\}$



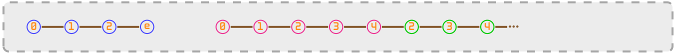
Compute an invariant on program variables at a program point

- To check the (un)reachability of unsafe states
- To optimize the code

```
@{ x:ℤ, y:ℤ }
1 x = 2;   @{ x:ℤ, y:ℤ }
2 while (y < x) {   @{ x:{2}, y:ℤ }
    3 y = y + 1;   @{ x:{2}, y:{z ∈ ℤ | z ≤ 1} }
    4 x = x   @{ x:{2}, y:{z ∈ ℤ | z ≤ 2} }
  }
@{ x:{2}, y:{z ∈ ℤ | z ≥ 2} }
```



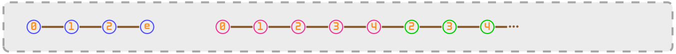
$$[\![P]\!] \in \wp(\mathbb{S}^* \cup \mathbb{S}^\infty)$$



$$[\![P]\!] \in \wp(\mathbb{S}^* \cup \mathbb{S}^\infty)$$

$$\Big\downarrow \alpha_\propto$$

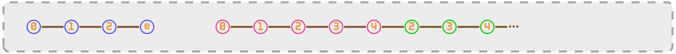
$$[\![P]\!]^\propto \in \wp(\mathbb{S}^*)$$



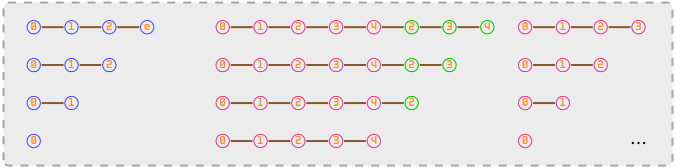
$$\llbracket P \rrbracket \in \wp(\mathbb{S}^* \cup \mathbb{S}^\infty)$$

$$\Big\downarrow \alpha_\propto$$

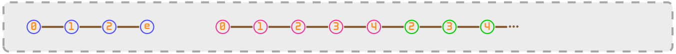
$$\llbracket P \rrbracket^\propto \in \wp(\mathbb{S}^*)$$



Trace prefixes



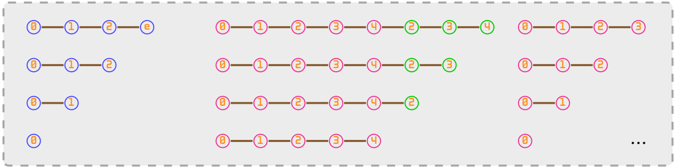
$$[\![P]\!] \in \wp(\mathbb{S}^* \cup \mathbb{S}^\infty)$$



$$\Big\downarrow \alpha_\propto$$

$$[\![P]\!]^\propto \in \wp(\mathbb{S}^*)$$

Trace prefixes



$$\Big\downarrow \alpha_\mathrm{r}$$

$$[\![P]\!]^\mathrm{r} \in \wp(\mathbb{S})$$

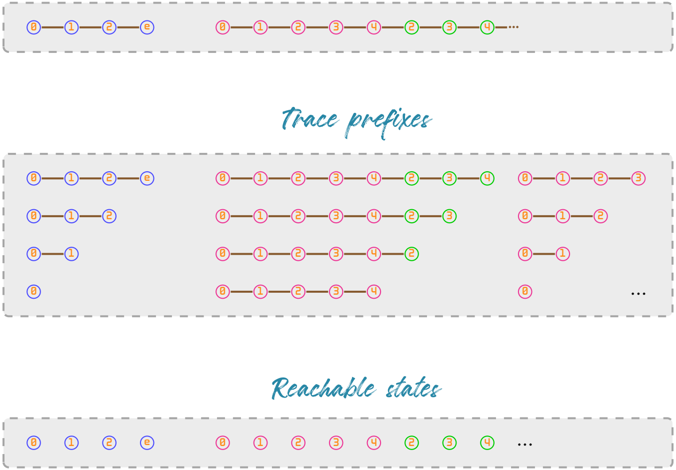
$[\![P]\!] \in \wp(\mathbb{S}^* \cup \mathbb{S}^\infty)$

$\downarrow \alpha_\propto$

$[\![P]\!]^\propto \in \wp(\mathbb{S}^*)$

$\downarrow \alpha_\mathrm{r}$

$[\![P]\!]^\mathrm{r} \in \wp(\mathbb{S})$



Reachability semantics: alternative definition

We can compute $[\![P]\!]^{\mathrm{r}}$ as least fixpoint of $\mathrm{f}_r : \wp(\mathbb{S}) \to \wp(\mathbb{S})$ inductively defined on P syntax

Reachability semantics: alternative definition

We can compute $[\![P]\!]^{\mathrm{r}}$ as least fixpoint of $\mathrm{f}_r : \wp(\mathbb{S}) \to \wp(\mathbb{S})$ inductively defined on P syntax

The sets $\wp(\mathbb{S}) = \wp(\mathbb{L} \times \mathbb{M})$ and $\mathbb{L} \to \wp(\mathbb{M})$ are isomorphic

We can compute $[\![P]\!]^{r}$ as least fixpoint of $\mathsf{f}_r : \wp(\mathbb{S}) \to \wp(\mathbb{S})$ inductively defined on P syntax

The sets $\wp(\mathbb{S}) = \wp(\mathbb{L} \times \mathbb{M})$ and $\mathbb{L} \to \wp(\mathbb{M})$ are isomorphic

Indeed, we have the Galois Correspondence

$$\langle \wp(\mathbb{L} \times \mathbb{M}), \subseteq \rangle \underset{\alpha_\ell}{\overset{\gamma_\ell}{\leftrightarrows}} \langle \mathbb{L} \to \wp(\mathbb{M}), \dot{\subseteq} \rangle$$

where

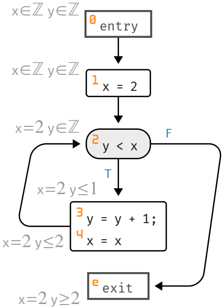
$$\alpha_\ell \triangleq \lambda X\,.\,\lambda \ell\,.\,\{\mathbb{m} \in \mathbb{M} \mid (\ell, \mathbb{m}) \in X\} \qquad \gamma_\ell \triangleq \lambda f\,.\,\{(\ell, \mathbb{m}) \in \mathbb{L} \times \mathbb{M} \mid \mathbb{m} \in f(\ell)\}$$

Reachability semantics: alternative definition (Cont'd)

In static analysis, we often use the alternative (yet isomorphic) representation $\alpha_\ell(\llbracket P \rrbracket^{\mathsf{r}})$

In static analysis, we often use the alternative (yet isomorphic) representation $\alpha_\ell(\llbracket P \rrbracket^{\mathrm{r}})$

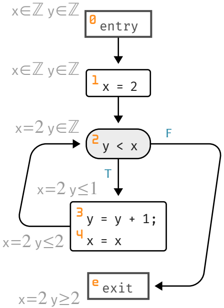
Change representation of reachability semantics: attach memory invariants to program points



In static analysis, we often use the alternative (yet isomorphic) representation $\alpha_\ell(\llbracket P \rrbracket^{\mathrm{r}})$

Change representation of reachability semantics: attach memory invariants to program points

- $\alpha_\ell(\llbracket P \rrbracket^{\mathrm{r}})(\ell)$ is the most precise (memory) invariant at point ℓ

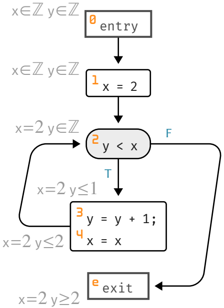


Reachability semantics: alternative definition (Cont'd)

In static analysis, we often use the alternative (yet isomorphic) representation $\alpha_\ell(\llbracket P \rrbracket^r)$

Change representation of reachability semantics: attach memory invariants to program points

- $\alpha_\ell(\llbracket P \rrbracket^r)(\ell)$ is the most precise (memory) invariant at point ℓ
- Standard settings of many static analyses (e.g., data-flow)



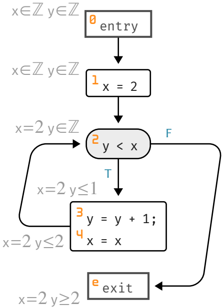
Reachability semantics: alternative definition (Cont'd)

In static analysis, we often use the alternative (yet isomorphic) representation $\alpha_\ell(\llbracket P \rrbracket^{\mathrm{r}})$

Change representation of reachability semantics: attach memory invariants to program points

- $\alpha_\ell(\llbracket P \rrbracket^{\mathrm{r}})(\ell)$ is the most precise (memory) invariant at point ℓ
- Standard settings of many static analyses (e.g., data-flow)
- Usually computed solving a system of equations:

$$(X_\ell = \mathbf{F}_\ell(X_0, \ldots, X_\ell, \ldots, X_e))_{\ell \in \{0,1,2,3,4,e\}}$$



Define transfer functions for conditions and code blocks

Conditions $\llparenthesis\, {}^{\ell} bexp \,\rrparenthesis : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

$\llparenthesis\, {}^{\ell} \mathtt{x} = exp \,\rrparenthesis : \wp(\mathbb{M}) \to \wp(\mathbb{M})$ Code blocks

Define transfer functions for conditions and code blocks

Conditions $\llparenthesis\, {}^{\ell}bexp \,\rrparenthesis : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

$\llparenthesis\, {}^{\ell}\texttt{x} = exp \,\rrparenthesis : \wp(\mathbb{M}) \to \wp(\mathbb{M})$ Code blocks

- $\llparenthesis\, {}^{\ell}bexp \,\rrparenthesis(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \mathsf{true}\}$

Define transfer functions for conditions and code blocks

Conditions $\llparenthesis\, {}^{\ell} bexp\, \rrparenthesis : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

$\llparenthesis\, {}^{\ell} \texttt{x= } exp\, \rrparenthesis : \wp(\mathbb{M}) \to \wp(\mathbb{M})$ Code blocks

- $\llparenthesis\, {}^{\ell} bexp\, \rrparenthesis(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \text{true}\}$

$\llparenthesis\, {}^{\ell} \texttt{x<y}\, \rrparenthesis(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[\texttt{x<y}](\mathbb{m}) = \text{true}\}$

Define transfer functions for conditions and code blocks

Conditions $(\!|{}^{\ell}bexp|\!) : \wp(\mathbb{M}) \to \wp(\mathbb{M})$ $(\!|{}^{\ell}\mathtt{x} = exp|\!) : \wp(\mathbb{M}) \to \wp(\mathbb{M})$ Code blocks

- $(\!|{}^{\ell}bexp|\!)(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \mathsf{true}\}$

$(\!|{}^{\ell}\mathtt{x<y}|\!)(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[\mathtt{x<y}](\mathbb{m}) = \mathsf{true}\} = \{\mathbb{m} \in X \mid \mathbb{m}(\mathtt{x}) < \mathbb{m}(\mathtt{y})\}$

Define transfer functions for conditions and code blocks

Conditions $(\!|\,{}^{\ell}bexp\,|\!) : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

$(\!|\,{}^{\ell}\mathtt{x} = exp\,|\!) : \wp(\mathbb{M}) \to \wp(\mathbb{M})$ Code blocks

- $(\!|\,{}^{\ell}bexp\,|\!)(X) \triangleq \{\mathbb{m} \in X \mid \text{EVAL}[bexp](\mathbb{m}) = \text{true}\}$

$$(\!|\,{}^{\ell}\mathtt{x{<}y}\,|\!)(X) \triangleq \{\mathbb{m} \in X \mid \text{EVAL}[\mathtt{x{<}y}](\mathbb{m}) = \text{true}\} = \{\mathbb{m} \in X \mid \mathbb{m}(\mathtt{x}) < \mathbb{m}(\mathtt{y})\}$$

- $(\!|\,{}^{\ell}\mathtt{x} = exp\,|\!)(X) \triangleq \{\mathbb{m}[\mathtt{x} \hookleftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \text{EVAL}[exp](\mathbb{m})\}$

Define transfer functions for conditions and code blocks

Conditions $\llparenthesis\, ^{\ell}bexp \,\rrparenthesis : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

$\llparenthesis\, ^{\ell}\texttt{x= } exp \,\rrparenthesis : \wp(\mathbb{M}) \to \wp(\mathbb{M})$ Code blocks

- $\llparenthesis\, ^{\ell}bexp \,\rrparenthesis(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \mathsf{true}\}$

$$\llparenthesis\, ^{\ell}\texttt{x<y} \,\rrparenthesis(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[\texttt{x<y}](\mathbb{m}) = \mathsf{true}\} = \{\mathbb{m} \in X \mid \mathbb{m}(\texttt{x}) < \mathbb{m}(\texttt{y})\}$$

- $\llparenthesis\, ^{\ell}\texttt{x= } exp \,\rrparenthesis(X) \triangleq \{\mathbb{m}[\texttt{x} \hookleftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[exp](\mathbb{m})\}$

$$\llparenthesis\, ^{\ell}\texttt{x=y+2} \,\rrparenthesis(X) \triangleq \{\mathbb{m}[\texttt{x} \hookleftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[\texttt{y+2}](\mathbb{m})\}$$

Define transfer functions for conditions and code blocks

Conditions $\llparenthesis\, {}^{\ell}bexp \,\rrparenthesis : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$ $\qquad$ $\llparenthesis\, {}^{\ell}\mathtt{x}\texttt{=}\, exp \,\rrparenthesis : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$ Code blocks

- $\llparenthesis\, {}^{\ell}bexp \,\rrparenthesis(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \mathsf{true}\}$

$$\llparenthesis\, {}^{\ell}\mathtt{x{<}y} \,\rrparenthesis(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[\mathtt{x{<}y}](\mathbb{m}) = \mathsf{true}\} = \{\mathbb{m} \in X \mid \mathbb{m}(\mathtt{x}) < \mathbb{m}(\mathtt{y})\}$$

- $\llparenthesis\, {}^{\ell}\mathtt{x}\texttt{=}\, exp \,\rrparenthesis(X) \triangleq \{\mathbb{m}[\mathtt{x} \hookleftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[exp](\mathbb{m})\}$

$$\llparenthesis\, {}^{\ell}\mathtt{x{=}y{+}2} \,\rrparenthesis(X) \triangleq \{\mathbb{m}[\mathtt{x} \hookleftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[\mathtt{y{+}2}](\mathbb{m})\} = \{\mathbb{m}[\mathtt{x} \hookleftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \mathbb{m}(\mathtt{y}) + 2\}$$

The reachability semantics is the least solution of the system of equations

$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{M} \\ X_{\ell} \triangleq \bigcup_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt \,|\!) X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

The reachability semantics is the least solution of the system of equations

$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{M} \\ X_{\ell} \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis X_{\ell'} \quad \text{if } \mathrm{next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

Each $X_{\ell} \mapsto \bigcup_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis X_{\ell'}$ is monotonic on the complete lattice $\langle \wp(\mathbb{M}), \subseteq, \cup, \cap, \emptyset, \mathbb{M} \rangle$

Equational reachability semantics (Cont'd)

The reachability semantics is the least solution of the system of equations

$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{M} \\ X_{\ell} \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis {}^{\ell'}stmt \rrparenthesis X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

Each $X_{\ell} \mapsto \bigcup_{\ell' \in \mathbb{L}} \llparenthesis {}^{\ell'}stmt \rrparenthesis X_{\ell'}$ is monotonic on the complete lattice $\langle \wp(\mathbb{M}), \subseteq, \cup, \cap, \emptyset, \mathbb{M} \rangle$

- The least fixpoint does exist

Equational reachability semantics (Cont'd)

The reachability semantics is the least solution of the system of equations

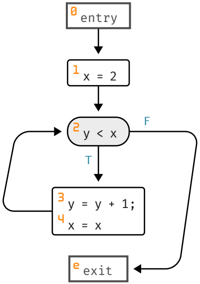
$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{M} \\ X_{\ell} \triangleq \bigcup_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!) X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

Each $X_\ell \mapsto \bigcup_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!) X_{\ell'}$ is monotonic on the complete lattice $\langle \wp(\mathbb{M}), \subseteq, \cup, \cap, \emptyset, \mathbb{M} \rangle$

- The least fixpoint does exist
- The solution can be computed by increasing iterations

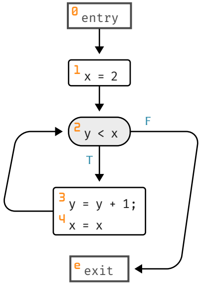
$$\begin{cases} X_{\mathtt{0}}^0 \triangleq \mathbb{M} \\ X_{\ell}^0 \triangleq \emptyset \end{cases} \qquad \begin{cases} X_{\mathtt{0}}^{n+1} \triangleq \mathbb{M} \\ X_{\ell}^{n+1} \triangleq \bigcup_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!) X_{\ell'}^{n} \end{cases}$$

Equational reachability semantics: example



$$\begin{cases} X_{0} \triangleq \mathbb{M} \\ X_{\ell} \triangleq \bigcup_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!) X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

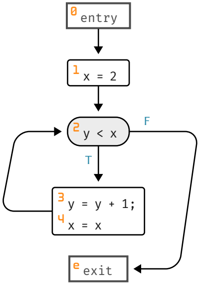
Equational reachability semantics: example



$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{M} \\ X_{\ell} \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'} stmt \,\rrparenthesis X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

$$X_{\mathtt{0}} = \mathbb{M}$$
$$X_{\mathtt{1}} = \llparenthesis\, {}^{\mathtt{0}}\texttt{entry} \,\rrparenthesis X_{\mathtt{0}}$$
$$X_{\mathtt{2}} = \llparenthesis\, {}^{\mathtt{1}}\texttt{x = 2} \,\rrparenthesis X_{\mathtt{1}} \cup \llparenthesis\, {}^{\mathtt{4}}\texttt{x = x} \,\rrparenthesis X_{\mathtt{4}}$$
$$X_{\mathtt{3}} = \llparenthesis\, {}^{\mathtt{2}}\texttt{y < x} \,\rrparenthesis X_{\mathtt{2}}$$
$$X_{\mathtt{4}} = \llparenthesis\, {}^{\mathtt{3}}\texttt{y = y + 1} \,\rrparenthesis X_{\mathtt{3}}$$
$$X_{\mathtt{e}} = \llparenthesis\, {}^{\mathtt{2}}\texttt{y >= x} \,\rrparenthesis X_{\mathtt{2}}$$

Equational reachability semantics: example

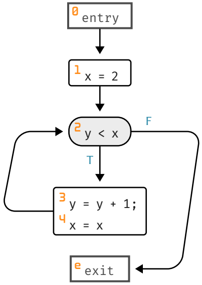


$$\begin{cases} X_0 \triangleq \mathbb{M} \\ X_\ell \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'} stmt \,\rrparenthesis X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

X_0	X_1	X_2	X_3	X_4	X_e

$$X_0 = \mathbb{M}$$
$$X_1 = \llparenthesis\, {}^{0}\texttt{entry} \,\rrparenthesis X_0$$
$$X_2 = \llparenthesis\, {}^{1}\texttt{x = 2} \,\rrparenthesis X_1 \cup \llparenthesis\, {}^{4}\texttt{x = x} \,\rrparenthesis X_4$$
$$X_3 = \llparenthesis\, {}^{2}\texttt{y < x} \,\rrparenthesis X_2$$
$$X_4 = \llparenthesis\, {}^{3}\texttt{y = y + 1} \,\rrparenthesis X_3$$
$$X_e = \llparenthesis\, {}^{2}\texttt{y >= x} \,\rrparenthesis X_2$$

Equational reachability semantics: example

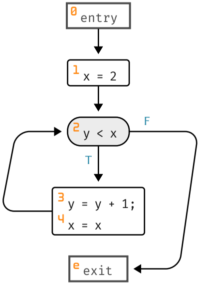


$$\begin{cases} X_0 \triangleq \mathbb{M} \\ X_\ell \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis {}^{\ell'} stmt \rrparenthesis X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

$$X_0 = \mathbb{M}$$
$$X_1 = \llparenthesis {}^{0}\texttt{entry} \rrparenthesis X_0$$
$$X_2 = \llparenthesis {}^{1}\texttt{x = 2} \rrparenthesis X_1 \cup \llparenthesis {}^{4}\texttt{x = x} \rrparenthesis X_4$$
$$X_3 = \llparenthesis {}^{2}\texttt{y < x} \rrparenthesis X_2$$
$$X_4 = \llparenthesis {}^{3}\texttt{y = y + 1} \rrparenthesis X_3$$
$$X_e = \llparenthesis {}^{2}\texttt{y >= x} \rrparenthesis X_2$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Equational reachability semantics: example



$$\begin{cases} X_{0} \triangleq \mathbb{M} \\ X_{\ell} \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

$$X_{0} = \mathbb{M}$$

$$X_{1} = \llparenthesis\, {}^{0}\texttt{entry} \,\rrparenthesis X_{0}$$

$$X_{2} = \llparenthesis\, {}^{1}\texttt{x = 2} \,\rrparenthesis X_{1} \cup \llparenthesis\, {}^{4}\texttt{x = x} \,\rrparenthesis X_{4}$$

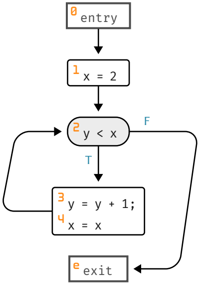
$$X_{3} = \llparenthesis\, {}^{2}\texttt{y < x} \,\rrparenthesis X_{2}$$

$$X_{4} = \llparenthesis\, {}^{3}\texttt{y = y + 1} \,\rrparenthesis X_{3}$$

$$X_{e} = \llparenthesis\, {}^{2}\texttt{y >= x} \,\rrparenthesis X_{2}$$

X_{0}	X_{1}	X_{2}	X_{3}	X_{4}	X_{e}
$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Equational reachability semantics: example

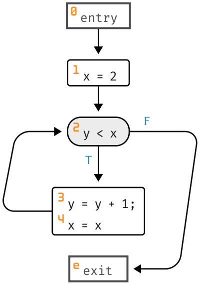


$$\begin{cases} X_0 \triangleq \mathbb{M} \\ X_\ell \triangleq \bigcup_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt |\!) X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

$$X_0 = \mathbb{M}$$
$$X_1 = (\!| {}^{0}\texttt{entry} |\!) X_0$$
$$X_2 = (\!| {}^{1}\texttt{x = 2} |\!) X_1 \cup (\!| {}^{4}\texttt{x = x} |\!) X_4$$
$$X_3 = (\!| {}^{2}\texttt{y < x} |\!) X_2$$
$$X_4 = (\!| {}^{3}\texttt{y = y + 1} |\!) X_3$$
$$X_e = (\!| {}^{2}\texttt{y >= x} |\!) X_2$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$(\{2\}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$

Equational reachability semantics: example

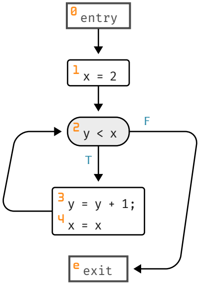


$$
\begin{cases}
X_0 \triangleq \mathbb{M} \\
X_\ell \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis {}^{\ell'}stmt \rrparenthesis X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell
\end{cases}
$$

$$
\begin{aligned}
X_0 &= \mathbb{M} \\
X_1 &= \llparenthesis {}^{0}\texttt{entry} \rrparenthesis X_0 \\
X_2 &= \llparenthesis {}^{1}\texttt{x = 2} \rrparenthesis X_1 \cup \llparenthesis {}^{4}\texttt{x = x} \rrparenthesis X_4 \\
X_3 &= \llparenthesis {}^{2}\texttt{y < x} \rrparenthesis X_2 \\
X_4 &= \llparenthesis {}^{3}\texttt{y = y + 1} \rrparenthesis X_3 \\
X_e &= \llparenthesis {}^{2}\texttt{y >= x} \rrparenthesis X_2
\end{aligned}
$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\mathbb{Z},\mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z},\mathbb{Z})$	$(\mathbb{Z},\mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z},\mathbb{Z})$	$(\mathbb{Z},\mathbb{Z})$	$(\{2\},\mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z},\mathbb{Z})$	$(\mathbb{Z},\mathbb{Z})$	$(\{2\},\mathbb{Z})$	$(\{2\},\mathbb{Z}^{<2})$	$\emptyset$	$\emptyset$

Equational reachability semantics: example

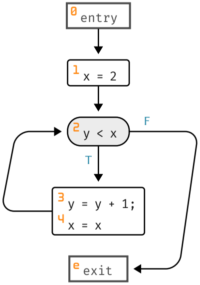


$$\begin{cases} X_0 \triangleq \mathbb{M} \\ X_\ell \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis {}^{\ell'}stmt \rrparenthesis X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

$$X_0 = \mathbb{M}$$
$$X_1 = \llparenthesis {}^{0}\texttt{entry} \rrparenthesis X_0$$
$$X_2 = \llparenthesis {}^{1}\texttt{x = 2} \rrparenthesis X_1 \cup \llparenthesis {}^{4}\texttt{x = x} \rrparenthesis X_4$$
$$X_3 = \llparenthesis {}^{2}\texttt{y < x} \rrparenthesis X_2$$
$$X_4 = \llparenthesis {}^{3}\texttt{y = y + 1} \rrparenthesis X_3$$
$$X_e = \llparenthesis {}^{2}\texttt{y >= x} \rrparenthesis X_2$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$(\{2\}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$(\{2\}, \mathbb{Z})$	$(\{2\}, \mathbb{Z}^{<2})$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$(\{2\}, \mathbb{Z})$	$(\{2\}, \mathbb{Z}^{<2})$	$(\{2\}, \mathbb{Z}^{<3})$	$\emptyset$

Equational reachability semantics: example



$$\begin{cases} X_0 \triangleq \mathbb{M} \\ X_\ell \triangleq \bigcup_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis X_{\ell'} \quad \text{if } \mathrm{next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

$$\begin{aligned}
X_0 &= \mathbb{M} \\
X_1 &= \llparenthesis\, {}^{0}\texttt{entry} \,\rrparenthesis X_0 \\
X_2 &= \llparenthesis\, {}^{1}\texttt{x = 2} \,\rrparenthesis X_1 \cup \llparenthesis\, {}^{4}\texttt{x = x} \,\rrparenthesis X_4 \\
X_3 &= \llparenthesis\, {}^{2}\texttt{y < x} \,\rrparenthesis X_2 \\
X_4 &= \llparenthesis\, {}^{3}\texttt{y = y + 1} \,\rrparenthesis X_3 \\
X_e &= \llparenthesis\, {}^{2}\texttt{y >= x} \,\rrparenthesis X_2
\end{aligned}$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$(\{2\}, \mathbb{Z})$	$\emptyset$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$(\{2\}, \mathbb{Z})$	$(\{2\}, \mathbb{Z}^{<2})$	$\emptyset$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$(\{2\}, \mathbb{Z})$	$(\{2\}, \mathbb{Z}^{<2})$	$(\{2\}, \mathbb{Z}^{<3})$	$\emptyset$
$(\mathbb{Z}, \mathbb{Z})$	$(\mathbb{Z}, \mathbb{Z})$	$(\{2\}, \mathbb{Z})$	$(\{2\}, \mathbb{Z}^{<2})$	$(\{2\}, \mathbb{Z}^{<3})$	$(\{2\}, \mathbb{Z}^{\geq 2})$

Compute the invariant on the last control point only

Compute the invariant on the last control point only

$$\langle \wp(\mathbb{S}), \subseteq \rangle \underset{\alpha_{\mathsf{e}}}{\overset{\gamma_{\mathsf{e}}}{\leftrightarrows}} \langle \wp(\mathbb{M}), \subseteq \rangle$$

$$\alpha_{\mathsf{e}} \triangleq \lambda X \,.\, \{\mathtt{m} \in \mathbb{M} \mid (\mathsf{e}, \mathtt{m}) \in X\} \qquad \gamma_{\mathsf{e}} \triangleq \lambda X \,.\, \{(\mathsf{e}, \mathtt{m}) \in \mathbb{L} \times \mathbb{M} \mid \mathtt{m} \in X\} \cup (\mathbb{L} \setminus \{\mathsf{e}\}) \times \mathbb{M}$$

Compute the invariant on the last control point only *// reduce memory space by a LoC factor*

$$\langle \wp(\mathbb{S}), \subseteq \rangle \underset{\alpha_{\mathsf{e}}}{\overset{\gamma_{\mathsf{e}}}{\leftrightarrows}} \langle \wp(\mathbb{M}), \subseteq \rangle$$

$$\alpha_{\mathsf{e}} \triangleq \lambda X \,.\, \{\mathbb{m} \in \mathbb{M} \mid (\mathsf{e}, \mathbb{m}) \in X\}$$

$$\gamma_{\mathsf{e}} \triangleq \lambda X \,.\, \{(\mathsf{e}, \mathbb{m}) \in \mathbb{L} \times \mathbb{M} \mid \mathbb{m} \in X\} \cup (\mathbb{L} \setminus \{\mathsf{e}\}) \times \mathbb{M}$$

Compute the invariant on the last control point only *// reduce memory space by a LoC factor*

$$\langle \wp(\mathbb{S}), \subseteq \rangle \underset{\alpha_{\mathsf{e}}}{\overset{\gamma_{\mathsf{e}}}{\leftrightarrows}} \langle \wp(\mathbb{M}), \subseteq \rangle$$

$\alpha_{\mathsf{e}} \triangleq \lambda X \,.\, \{ \mathfrak{m} \in \mathbb{M} \mid (\mathsf{e}, \mathfrak{m}) \in X \}$ $\gamma_{\mathsf{e}} \triangleq \lambda X \,.\, \{ (\mathsf{e}, \mathfrak{m}) \in \mathbb{L} \times \mathbb{M} \mid \mathfrak{m} \in X \} \cup (\mathbb{L} \setminus \{\mathsf{e}\}) \times \mathbb{M}$

Post-condition semantics $[\![P]\!]^{\mathrm{c}} = \alpha_{\mathsf{e}}([\![P]\!]^{\mathrm{r}})$

Compute the invariant on the last control point only // *reduce memory space by a LoC factor*

$$\langle \wp(\mathbb{S}), \subseteq \rangle \xleftrightarrow[\alpha_{\mathsf{e}}]{\gamma_{\mathsf{e}}} \langle \wp(\mathbb{M}), \subseteq \rangle$$

$\alpha_{\mathsf{e}} \triangleq \lambda X . \{ \mathbb{m} \in \mathbb{M} \mid (\mathsf{e}, \mathbb{m}) \in X \}$ $\gamma_{\mathsf{e}} \triangleq \lambda X . \{ (\mathsf{e}, \mathbb{m}) \in \mathbb{L} \times \mathbb{M} \mid \mathbb{m} \in X \} \cup (\mathbb{L} \setminus \{\mathsf{e}\}) \times \mathbb{M}$

Post-condition semantics $[\![P]\!]^{\mathrm{c}} = \alpha_{\mathsf{e}}([\![P]\!]^{\mathrm{r}})$

- Inductively computed on program syntax $[\![P]\!]^{\mathrm{c}} \triangleq \mathbf{S}^{\mathrm{c}}_{stmt_n} \circ \ldots \circ \mathbf{S}^{\mathrm{c}}_{stmt_1}(\mathfrak{I})$ // $\mathfrak{I} \subseteq \mathbb{M}$
- Given a transfer function $\mathbf{S}^{\mathrm{c}} : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$ for all statements $stmt_1, \ldots, stmt_n$ of P

Compute the invariant on the last control point only // *reduce memory space by a LoC factor*

$$\langle \wp(\mathbb{S}), \subseteq \rangle \underset{\alpha_{e}}{\overset{\gamma_{e}}{\leftrightarrows}} \langle \wp(\mathbb{M}), \subseteq \rangle$$

$\alpha_{e} \triangleq \lambda X.\ \{\mathsf{m} \in \mathbb{M} \mid (e, \mathsf{m}) \in X\}$ $\gamma_{e} \triangleq \lambda X.\ \{(e, \mathsf{m}) \in \mathbb{L} \times \mathbb{M} \mid \mathsf{m} \in X\} \cup (\mathbb{L} \setminus \{e\}) \times \mathbb{M}$

Define $\langle \mathbb{L} \to \wp(\mathbb{M}), \dot{\subseteq} \rangle \underset{\bar{\alpha}_{e}}{\overset{\bar{\gamma}_{e}}{\leftrightarrows}} \langle \wp(\mathbb{M}), \subseteq \rangle$

Post-condition semantics $[\![P]\!]^{c} = \alpha_{e}([\![P]\!]^{r})$

- Inductively computed on program syntax $[\![P]\!]^{c} \triangleq \mathbf{S}^{c}_{stmt_n} \circ \ldots \circ \mathbf{S}^{c}_{stmt_1}(\mathfrak{I})$ // $\mathfrak{I} \subseteq \mathbb{M}$
- Given a transfer function $\mathbf{S}^{c} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$ for all statements $stmt_1, \ldots, stmt_n$ of P

Boolean expressions $\mathbf{S}^{c}_{bexp} : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$

Programs $\mathbf{S}^{c}_{P} : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$

Boolean expressions $\mathbf{S}^{c}_{bexp} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{bexp}(X) \triangleq \{\mathfrak{m} \in X \mid \textsc{Eval}[bexp](\mathfrak{m}) = \mathsf{true}\}$

Programs $\mathbf{S}^{c}_{P} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

Post-condition semantics (Cont'd)

Boolean expressions $\mathbf{S}^{c}_{bexp} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{bexp}(X) \triangleq \{\mathfrak{m} \in X \mid \textsc{Eval}[bexp](\mathfrak{m}) = \mathsf{true}\}$

$$\mathbf{S}^{c}_{\mathsf{x<y}}(X) \triangleq \{\mathfrak{m} \in X \mid \textsc{Eval}[\mathsf{x<y}](\mathfrak{m}) = \mathsf{true}\}$$

Programs $\mathbf{S}^{c}_{P} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

Post-condition semantics (Cont'd)

Boolean expressions $\mathbf{S}^{c}_{bexp} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{bexp}(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \mathsf{true}\}$

$$\mathbf{S}^{c}_{\mathsf{x<y}}(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[\mathsf{x<y}](\mathbb{m}) = \mathsf{true}\} = \{\mathbb{m} \in X \mid \mathbb{m}(\mathsf{x}) < \mathbb{m}(\mathsf{y})\}$$

Programs $\mathbf{S}^{c}_{P} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

Post-condition semantics (Cont'd)

Boolean expressions $\mathbf{S}^{c}_{bexp} : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{bexp}(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \mathsf{true}\}$

$$\mathbf{S}^{c}_{\mathtt{x<y}}(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[\mathtt{x<y}](\mathbb{m}) = \mathsf{true}\} = \{\mathbb{m} \in X \mid \mathbb{m}(\mathtt{x}) < \mathbb{m}(\mathtt{y})\}$$

Programs $\mathbf{S}^{c}_{P} : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{\mathtt{x}=exp}(X) \triangleq \{\mathbb{m}[\mathtt{x} \leftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[exp](\mathbb{m})\}$

Post-condition semantics (Cont'd)

Boolean expressions $\mathbf{S}^{c}_{bexp} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{bexp}(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \mathsf{true}\}$

$$\mathbf{S}^{c}_{\mathtt{x<y}}(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[\mathtt{x<y}](\mathbb{m}) = \mathsf{true}\} = \{\mathbb{m} \in X \mid \mathbb{m}(\mathtt{x}) < \mathbb{m}(\mathtt{y})\}$$

Programs $\mathbf{S}^{c}_{P} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{\mathtt{x}=exp}(X) \triangleq \{\mathbb{m}[\mathtt{x} \leftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[exp](\mathbb{m})\}$

$$\mathbf{S}^{c}_{\mathtt{x=y+2}}(X) \triangleq \{\mathbb{m}[\mathtt{x} \leftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[\mathtt{y+2}](\mathbb{m})\}$$

Boolean expressions $\mathbf{S}^{c}_{bexp} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{bexp}(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[bexp](\mathbb{m}) = \mathsf{true}\}$

$$\mathbf{S}^{c}_{\mathtt{x<y}}(X) \triangleq \{\mathbb{m} \in X \mid \textsc{Eval}[\mathtt{x<y}](\mathbb{m}) = \mathsf{true}\} = \{\mathbb{m} \in X \mid \mathbb{m}(\mathtt{x}) < \mathbb{m}(\mathtt{y})\}$$

Programs $\mathbf{S}^{c}_{P} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{\mathtt{x}=exp}(X) \triangleq \{\mathbb{m}[\mathtt{x} \leftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[exp](\mathbb{m})\}$

$$\mathbf{S}^{c}_{\mathtt{x=y+2}}(X) \triangleq \{\mathbb{m}[\mathtt{x} \leftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \textsc{Eval}[\mathtt{y+2}](\mathbb{m})\} = \{\mathbb{m}[\mathtt{x} \leftarrow \mathbb{v}] \mid \mathbb{m} \in X \wedge \mathbb{v} = \mathbb{m}(\mathtt{y}) + 2\}$$

Boolean expressions $\mathbf{S}^{c}_{bexp} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{bexp}(X) \triangleq \{\mathrm{m} \in X \mid \textsc{Eval}[bexp](\mathrm{m}) = \mathsf{true}\}$

$$\mathbf{S}^{c}_{\mathtt{x<y}}(X) \triangleq \{\mathrm{m} \in X \mid \textsc{Eval}[\mathtt{x<y}](\mathrm{m}) = \mathsf{true}\} = \{\mathrm{m} \in X \mid \mathrm{m}(\mathtt{x}) < \mathrm{m}(\mathtt{y})\}$$

Programs $\mathbf{S}^{c}_{P} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^{c}_{\mathtt{x=}exp}(X) \triangleq \{\mathrm{m}[\mathtt{x} \leftarrow \mathrm{v}] \mid \mathrm{m} \in X \land \mathrm{v} = \textsc{Eval}[exp](\mathrm{m})\}$

$$\mathbf{S}^{c}_{\mathtt{x=y+2}}(X) \triangleq \{\mathrm{m}[\mathtt{x} \leftarrow \mathrm{v}] \mid \mathrm{m} \in X \land \mathrm{v} = \textsc{Eval}[\mathtt{y+2}](\mathrm{m})\} = \{\mathrm{m}[\mathtt{x} \leftarrow \mathrm{v}] \mid \mathrm{m} \in X \land \mathrm{v} = \mathrm{m}(\mathtt{y}) + 2\}$$

- $\mathbf{S}^{c}_{stmt_1\,;\,stmt_2}(X) \triangleq \mathbf{S}^{c}_{stmt_2} \circ \mathbf{S}^{c}_{stmt_1}(X)$

Boolean expressions $\mathbf{S}^c_{bexp} : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^c_{bexp}(X) \triangleq \{\mathfrak{m} \in X \mid \textsc{Eval}[bexp](\mathfrak{m}) = \mathsf{true}\}$

$$\mathbf{S}^c_{\mathtt{x<y}}(X) \triangleq \{\mathfrak{m} \in X \mid \textsc{Eval}[\mathtt{x<y}](\mathfrak{m}) = \mathsf{true}\} = \{\mathfrak{m} \in X \mid \mathfrak{m}(\mathtt{x}) < \mathfrak{m}(\mathtt{y})\}$$

Programs $\mathbf{S}^c_P : \wp(\mathbb{M}) \to \wp(\mathbb{M})$

- $\mathbf{S}^c_{\mathtt{x=}exp}(X) \triangleq \{\mathfrak{m}[\mathtt{x} \hookleftarrow \mathbb{v}] \mid \mathfrak{m} \in X \wedge \mathbb{v} = \textsc{Eval}[exp](\mathfrak{m})\}$

$$\mathbf{S}^c_{\mathtt{x=y+2}}(X) \triangleq \{\mathfrak{m}[\mathtt{x} \hookleftarrow \mathbb{v}] \mid \mathfrak{m} \in X \wedge \mathbb{v} = \textsc{Eval}[\mathtt{y+2}](\mathfrak{m})\} = \{\mathfrak{m}[\mathtt{x} \hookleftarrow \mathbb{v}] \mid \mathfrak{m} \in X \wedge \mathbb{v} = \mathfrak{m}(\mathtt{y}) + 2\}$$

- $\mathbf{S}^c_{stmt_1\,;\,stmt_2}(X) \triangleq \mathbf{S}^c_{stmt_2} \circ \mathbf{S}^c_{stmt_1}(X)$
- $\mathbf{S}^c_{\mathtt{if}(bexp)\{stmt_1\}\mathtt{else}\{stmt_2\}}(X) \triangleq \mathbf{S}^c_{stmt_1} \circ \mathbf{S}^c_{bexp}(X) \cup \mathbf{S}^c_{stmt_2} \circ \mathbf{S}^c_{\neg bexp}(X)$

Post-condition semantics (Cont'd)

Boolean expressions $\mathbf{S}^c_{bexp} : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$

- $\mathbf{S}^c_{bexp}(X) \triangleq \{\mathfrak{m} \in X \mid \text{EVAL}[bexp](\mathfrak{m}) = \text{true}\}$

$$\mathbf{S}^c_{\mathtt{x<y}}(X) \triangleq \{\mathfrak{m} \in X \mid \text{EVAL}[\mathtt{x<y}](\mathfrak{m}) = \text{true}\} = \{\mathfrak{m} \in X \mid \mathfrak{m}(\mathtt{x}) < \mathfrak{m}(\mathtt{y})\}$$

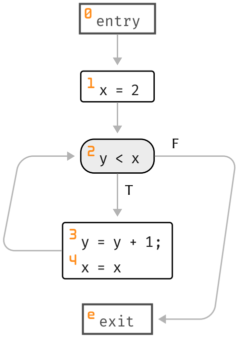
Programs $\mathbf{S}^c_{P} : \wp(\mathbb{M}) \rightarrow \wp(\mathbb{M})$

- $\mathbf{S}^c_{\mathtt{x=}exp}(X) \triangleq \{\mathfrak{m}[\mathtt{x} \leftarrow \mathfrak{v}] \mid \mathfrak{m} \in X \wedge \mathfrak{v} = \text{EVAL}[exp](\mathfrak{m})\}$

$$\mathbf{S}^c_{\mathtt{x=y+2}}(X) \triangleq \{\mathfrak{m}[\mathtt{x} \leftarrow \mathfrak{v}] \mid \mathfrak{m} \in X \wedge \mathfrak{v} = \text{EVAL}[\mathtt{y+2}](\mathfrak{m})\} = \{\mathfrak{m}[\mathtt{x} \leftarrow \mathfrak{v}] \mid \mathfrak{m} \in X \wedge \mathfrak{v} = \mathfrak{m}(\mathtt{y}) + 2\}$$

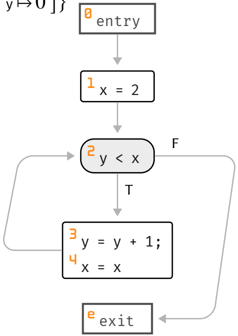
- $\mathbf{S}^c_{stmt_1\,;\,stmt_2}(X) \triangleq \mathbf{S}^c_{stmt_2} \circ \mathbf{S}^c_{stmt_1}(X)$
- $\mathbf{S}^c_{\mathtt{if}(bexp)\{stmt_1\}\mathtt{else}\{stmt_2\}}(X) \triangleq \mathbf{S}^c_{stmt_1} \circ \mathbf{S}^c_{bexp}(X) \cup \mathbf{S}^c_{stmt_2} \circ \mathbf{S}^c_{\neg bexp}(X)$
- $\mathbf{S}^c_{\mathtt{while}(bexp)\{stmt\}}(X) \triangleq \mathbf{S}^c_{\neg bexp} \circ (\text{lfp}^{\subseteq}\, \lambda Y\,.\, X \cup \mathbf{S}^c_{stmt} \circ \mathbf{S}^c_{bexp}(Y))$

Post-condition semantics: example



Post-condition semantics: example

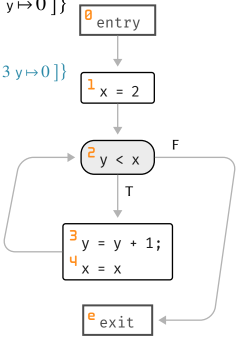
Input: $\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 3\ y \mapsto 0\,]\}$



Post-condition semantics: example

Input: $\{[\,x \mapsto 2 \;\, y \mapsto 3\,], [\,x \mapsto 3 \;\, y \mapsto 0\,]\}$

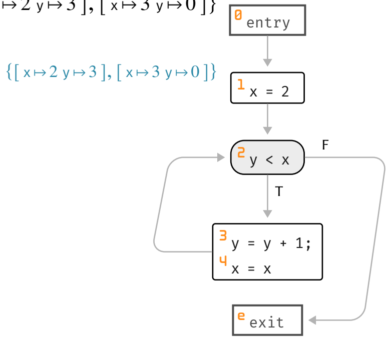
$\{[\,x \mapsto 2 \;\, y \mapsto 3\,], [\,x \mapsto 3 \;\, y \mapsto 0\,]\}$



Post-condition semantics: example

$$\mathbf{S}^{c}_{\texttt{x=2}}(X) = \{\mathbb{m}[\texttt{x} \hookleftarrow 2] \mid \mathbb{m} \in X\}$$

Input: $\{[\,\texttt{x} \mapsto 2\ \texttt{y} \mapsto 3\,], [\,\texttt{x} \mapsto 3\ \texttt{y} \mapsto 0\,]\}$



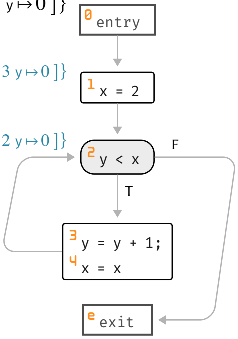
Post-condition semantics: example

$$\mathbf{S}^{c}_{\mathtt{x=2}}(X) = \{\mathbb{m}[\mathtt{x} \hookleftarrow 2] \mid \mathbb{m} \in X\} = \{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 0\,]\}$$

Input: $\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 3\ \mathtt{y}\mapsto 0\,]\}$

$\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 3\ \mathtt{y}\mapsto 0\,]\}$

$\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 0\,]\}$



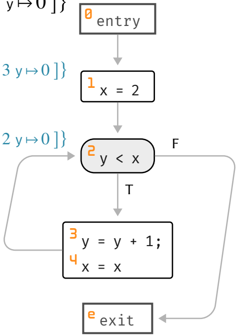
Post-condition semantics: example

$$\mathbf{S}^c_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^c_{\texttt{y>=x}}(\mathsf{lfp}^{\subseteq}\, \lambda Y\,.\, X \cup \mathbf{S}^c_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^c_{\texttt{y<x}}(Y))$$

Input: $\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 3\ \mathtt{y}\mapsto 0\,]\}$

$\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 3\ \mathtt{y}\mapsto 0\,]\}$

$\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 0\,]\}$



Post-condition semantics: example

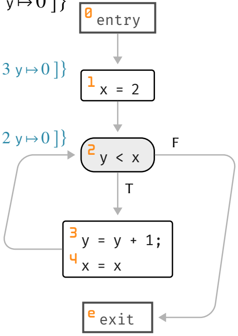
UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

$$\mathbf{S}^{c}_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^{c}_{\texttt{y>=x}}(\mathrm{lfp}^{\subseteq}\ \lambda Y\,.\ X \cup \mathbf{S}^{c}_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^{c}_{\texttt{y<x}}(Y))$$

Input: $\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 3\ y \mapsto 0\,]\}$

$\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 3\ y \mapsto 0\,]\}$

$\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\}$

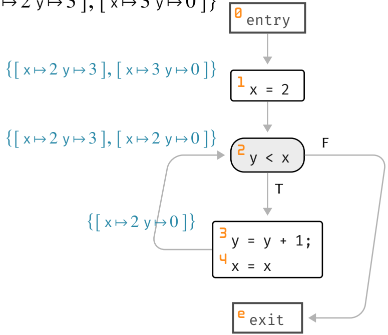


$$X^0 = \{\ [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\ \}$$

Post-condition semantics: example

$$\mathbf{S}^c_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^c_{\texttt{y>=x}}\left(\mathsf{lfp}^{\subseteq}\ \lambda Y\,.\ X \cup \mathbf{S}^c_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^c_{\texttt{y<x}}(Y)\right)$$

Input: $\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 3\ \mathtt{y}\mapsto 0\,]\}$

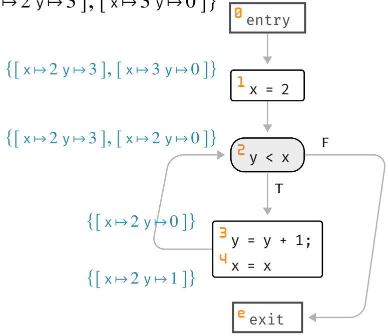


$$X^0 = \{\,[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 0\,]\,\}$$

Post-condition semantics: example

$$\mathbf{S}^c_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^c_{\texttt{y>=x}}\left(\mathsf{lfp}^{\subseteq}\ \lambda Y\,.\ X \cup \mathbf{S}^c_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^c_{\texttt{y<x}}(Y)\right)$$

Input: $\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 3\ y \mapsto 0\,]\}$

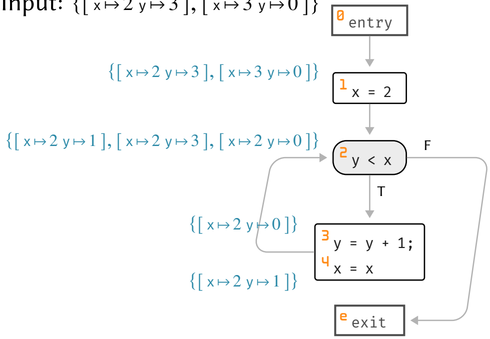


$$X^0 = \{\,[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\,\}$$

Post-condition semantics: example

$$\mathbf{S}^{c}_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^{c}_{\texttt{y>=x}}\left(\mathrm{lfp}^{\subseteq}\, \lambda Y\,.\, X \cup \mathbf{S}^{c}_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^{c}_{\texttt{y<x}}(Y)\right)$$

Input: $\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 3\ \mathtt{y}\mapsto 0\,]\}$



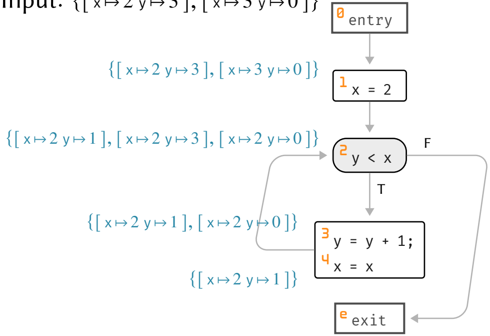
$$X^0 = \{\, [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 0\,] \,\}$$

$$X^1 = \left\{ \begin{array}{l} [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 1\,], [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], \\ [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 0\,] \end{array} \right\}$$

Post-condition semantics: example

$$\mathbf{S}^c_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^c_{\texttt{y>=x}}(\mathsf{lfp}^{\subseteq}\ \lambda Y\ .\ X \cup \mathbf{S}^c_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^c_{\texttt{y<x}}(Y))$$

Input: $\{[\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 3\ \mathtt{y}\mapsto 0\,]\}$



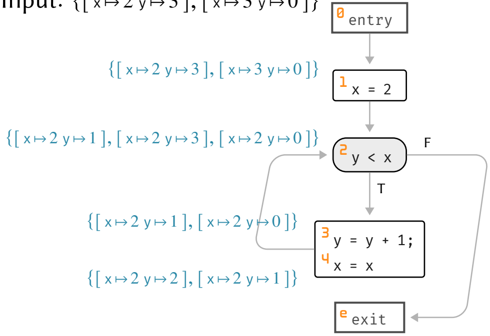
$$X^0 = \{\ [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 0\,]\ \}$$

$$X^1 = \left\{ \begin{array}{l} [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 1\,], [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 3\,], \\ [\,\mathtt{x}\mapsto 2\ \mathtt{y}\mapsto 0\,] \end{array} \right\}$$

Post-condition semantics: example

$$\mathbf{S}^{c}_{\texttt{while}}\texttt{(y<x)\{y=y+1;x=x\}}(X) = \mathbf{S}^{c}_{\texttt{y>=x}}\left(\mathsf{lfp}^{\subseteq}\ \lambda Y\,.\ X \cup \mathbf{S}^{c}_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^{c}_{\texttt{y<x}}(Y)\right)$$

Input: $\{[\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 3\,], [\,\mathsf{x}\mapsto 3\ \mathsf{y}\mapsto 0\,]\}$



$$X^0 = \{\, [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 3\,], [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 0\,] \,\}$$

$$X^1 = \left\{ \begin{array}{l} [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 1\,], [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 3\,], \\ [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 0\,] \end{array} \right\}$$

Post-condition semantics: example

$$\mathbf{S}^{c}_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^{c}_{\texttt{y>=x}}\left(\mathrm{lfp}^{\subseteq}\ \lambda Y.\ X \cup \mathbf{S}^{c}_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^{c}_{\texttt{y<x}}(Y)\right)$$

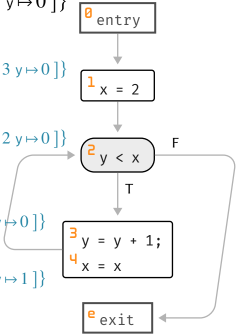
Input: $\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 3\ y \mapsto 0\,]\}$

$\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 3\ y \mapsto 0\,]\}$

$\{[\,x \mapsto 2\ y \mapsto 2\,], [\,x \mapsto 2\ y \mapsto 1\,], [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\}$

$\{[\,x \mapsto 2\ y \mapsto 1\,], [\,x \mapsto 2\ y \mapsto 0\,]\}$

$\{[\,x \mapsto 2\ y \mapsto 2\,], [\,x \mapsto 2\ y \mapsto 1\,]\}$



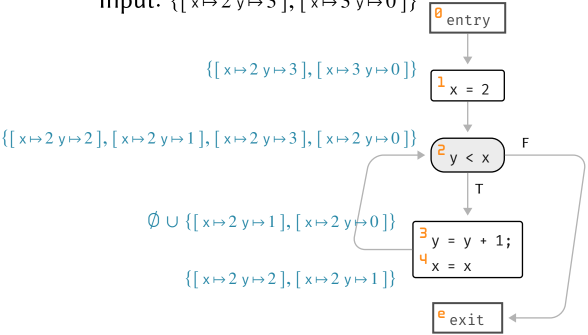
$$X^0 = \{\ [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\ \}$$

$$X^1 = \left\{ \begin{array}{l} [\,x \mapsto 2\ y \mapsto 1\,], [\,x \mapsto 2\ y \mapsto 3\,], \\ [\,x \mapsto 2\ y \mapsto 0\,] \end{array} \right\}$$

$$X^2 = \left\{ \begin{array}{l} [\,x \mapsto 2\ y \mapsto 2\,], [\,x \mapsto 2\ y \mapsto 1\,], \\ [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,] \end{array} \right\}$$

$$\mathbf{S}^c_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^c_{\texttt{y>=x}}(\mathsf{lfp}^{\subseteq}\ \lambda Y\,.\, X \cup \mathbf{S}^c_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^c_{\texttt{y<x}}(Y))$$

Input: $\{[\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 3\ y\mapsto 0\,]\}$



$$X^0 = \{\, [\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 2\ y\mapsto 0\,] \,\}$$

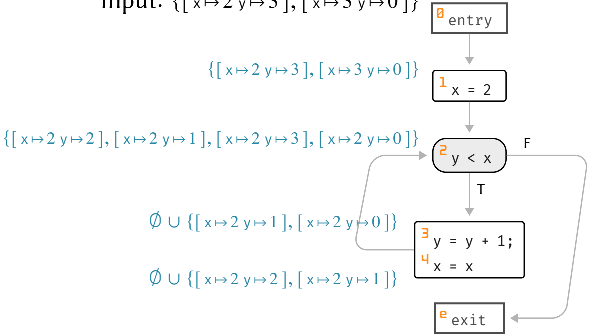
$$X^1 = \left\{ \begin{array}{l} [\,x\mapsto 2\ y\mapsto 1\,], [\,x\mapsto 2\ y\mapsto 3\,], \\ [\,x\mapsto 2\ y\mapsto 0\,] \end{array} \right\}$$

$$X^2 = \left\{ \begin{array}{l} [\,x\mapsto 2\ y\mapsto 2\,], [\,x\mapsto 2\ y\mapsto 1\,], \\ [\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 2\ y\mapsto 0\,] \end{array} \right\}$$

Post-condition semantics: example

$$\mathbf{S}^{c}_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^{c}_{\texttt{y>=x}}\left(\mathsf{lfp}^{\subseteq}\ \lambda Y\,.\ X \cup \mathbf{S}^{c}_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^{c}_{\texttt{y<x}}(Y)\right)$$

Input: $\{[\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 3\,], [\,\mathsf{x}\mapsto 3\ \mathsf{y}\mapsto 0\,]\}$



$$X^0 = \{\,[\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 3\,], [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 0\,]\,\}$$

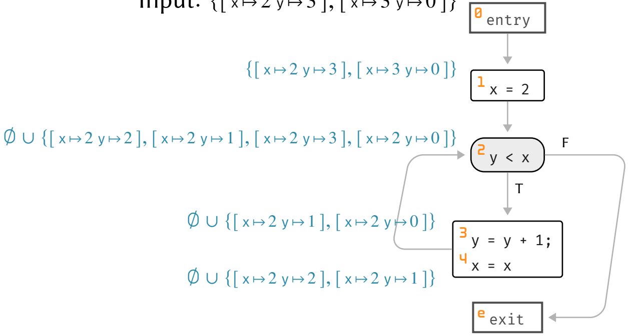
$$X^1 = \left\{\begin{array}{l}[\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 1\,], [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 3\,],\\ [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 0\,]\end{array}\right\}$$

$$X^2 = \left\{\begin{array}{l}[\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 2\,], [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 1\,],\\ [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 3\,], [\,\mathsf{x}\mapsto 2\ \mathsf{y}\mapsto 0\,]\end{array}\right\}$$

Post-condition semantics: example

$$\mathbf{S}^{c}_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^{c}_{\texttt{y>=x}}\big(\mathsf{lfp}^{\subseteq}\ \lambda Y\,.\ X \cup \mathbf{S}^{c}_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^{c}_{\texttt{y<x}}(Y)\big)$$

Input: $\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 3\ y \mapsto 0\,]\}$



$$X^0 = \{\ [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\ \}$$

$$X^1 = \left\{ \begin{array}{l} [\,x \mapsto 2\ y \mapsto 1\,], [\,x \mapsto 2\ y \mapsto 3\,], \\ [\,x \mapsto 2\ y \mapsto 0\,] \end{array} \right\}$$

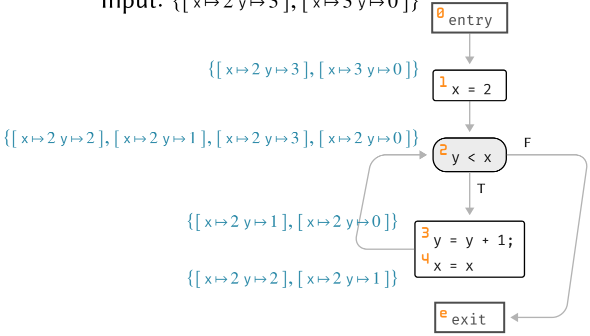
$$X^2 = \left\{ \begin{array}{l} [\,x \mapsto 2\ y \mapsto 2\,], [\,x \mapsto 2\ y \mapsto 1\,], \\ [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,] \end{array} \right\}$$

$$X^3 = \left\{ \begin{array}{l} [\,x \mapsto 2\ y \mapsto 2\,], [\,x \mapsto 2\ y \mapsto 1\,], \\ [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,] \end{array} \right\}$$

Post-condition semantics: example

$$\mathbf{S}^{c}_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^{c}_{\texttt{y>=x}}\left(\mathrm{lfp}^{\subseteq}\ \lambda Y\,.\ X \cup \mathbf{S}^{c}_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^{c}_{\texttt{y<x}}(Y)\right)$$

Input: $\{[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 3\ y \mapsto 0\,]\}$



$$X^0 = \{\,[\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\,\}$$

$$X^1 = \left\{\begin{array}{l}[\,x \mapsto 2\ y \mapsto 1\,], [\,x \mapsto 2\ y \mapsto 3\,],\\ [\,x \mapsto 2\ y \mapsto 0\,]\end{array}\right\}$$

$$X^2 = \left\{\begin{array}{l}[\,x \mapsto 2\ y \mapsto 2\,], [\,x \mapsto 2\ y \mapsto 1\,],\\ [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\end{array}\right\}$$

$$\mathrm{lfp} = \left\{\begin{array}{l}[\,x \mapsto 2\ y \mapsto 2\,], [\,x \mapsto 2\ y \mapsto 1\,],\\ [\,x \mapsto 2\ y \mapsto 3\,], [\,x \mapsto 2\ y \mapsto 0\,]\end{array}\right\}$$

Post-condition semantics: example

$$\mathbf{S}^c_{\texttt{while(y<x)\{y=y+1;x=x\}}}(X) = \mathbf{S}^c_{\texttt{y>=x}}\big(\mathsf{lfp}^{\subseteq}\ \lambda Y\,.\ X \cup \mathbf{S}^c_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^c_{\texttt{y<x}}(Y)\big)$$

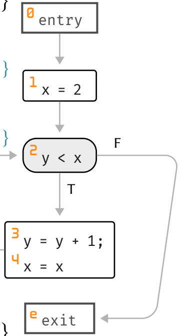
Input: $\{[\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 3\ y\mapsto 0\,]\}$

$\{[\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 3\ y\mapsto 0\,]\}$

$\{[\,x\mapsto 2\ y\mapsto 2\,], [\,x\mapsto 2\ y\mapsto 1\,], [\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 2\ y\mapsto 0\,]\}$

$\{[\,x\mapsto 2\ y\mapsto 1\,], [\,x\mapsto 2\ y\mapsto 0\,]\}$

$\{[\,x\mapsto 2\ y\mapsto 2\,], [\,x\mapsto 2\ y\mapsto 1\,]\}$



Output: $\{[\,x\mapsto 2\ y\mapsto 2\,], [\,x\mapsto 2\ y\mapsto 3\,]\}$

$$X^0 = \{\ [\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 2\ y\mapsto 0\,]\ \}$$

$$X^1 = \left\{\begin{array}{l}[\,x\mapsto 2\ y\mapsto 1\,], [\,x\mapsto 2\ y\mapsto 3\,],\\ [\,x\mapsto 2\ y\mapsto 0\,]\end{array}\right\}$$

$$X^2 = \left\{\begin{array}{l}[\,x\mapsto 2\ y\mapsto 2\,], [\,x\mapsto 2\ y\mapsto 1\,],\\ [\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 2\ y\mapsto 0\,]\end{array}\right\}$$

$$\mathsf{lfp} = \left\{\begin{array}{l}[\,x\mapsto 2\ y\mapsto 2\,], [\,x\mapsto 2\ y\mapsto 1\,],\\ [\,x\mapsto 2\ y\mapsto 3\,], [\,x\mapsto 2\ y\mapsto 0\,]\end{array}\right\}$$

Computing Abstract Invariants

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

Computing the best memory invariant is undecidable

Computing the best memory invariant is undecidable

- Elements in $\wp(\mathbb{M})$ are not computer-representable
- The transfer functions $(\!|\ {}^{\ell'} stmt \ |\!)$ and $\mathbf{S}^{c}_{stmt}$ are not computable
- The fixpoint iterations on $\wp(\mathbb{M})$ are transfinite, in general

Computing the best memory invariant is undecidable

- Elements in $\wp(\mathbb{M})$ are not computer-representable
- The transfer functions $(\!|\, {}^{\ell'} stmt \,|\!)$ and $\mathbf{S}^{c}_{stmt}$ are not computable
- The fixpoint iterations on $\wp(\mathbb{M})$ are transfinite, in general

Note: being $\mathbb{V}$ finite is not particularly helpful from a practical point of view

Computing the best memory invariant is undecidable

- Elements in $\wp(\mathbb{M})$ are not computer-representable
- The transfer functions $\llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis$ and $\mathbf{S}^{c}_{stmt}$ are not computable
- The fixpoint iterations on $\wp(\mathbb{M})$ are transfinite, in general

Note: being $\mathbb{V}$ finite is not particularly helpful from a practical point of view

- Representing elements in $\wp(\mathbb{X} \to \mathbb{V})$ in extension is expensive
- Explicitly computing $\llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis$ and $\mathbf{S}^{c}_{stmt}$ is expensive
- The lattice $\langle \wp(\mathbb{X} \to \mathbb{V}), \subseteq \rangle$ has large height, hence iterations are expensive

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$$

Abstract domain

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$$

Abstract domain

- The abstract domain $\mathbb{M}^\sharp$ must be machine-representable

$$\langle \wp(\mathbb{M}), \subseteq \rangle \xleftrightarrow[\alpha]{\gamma} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$$

Abstract domain

- The abstract domain $\mathbb{M}^\sharp$ must be machine-representable
- The abstract test $\mathbb{m}_1^\sharp \subseteq^\sharp \mathbb{m}_2^\sharp$ must be decidable

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$$

Abstract domain

- The abstract domain $\mathbb{M}^\sharp$ must be machine-representable
- The abstract test $\mathbb{m}_1^\sharp \subseteq^\sharp \mathbb{m}_2^\sharp$ must be decidable
- The (binary) join $\mathbb{m}_1^\sharp \cup^\sharp \mathbb{m}_2^\sharp$ must be computable

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$$

Abstract domain

- The abstract domain $\mathbb{M}^\sharp$ must be machine-representable
- The abstract test $\mathbb{m}_1^\sharp \subseteq^\sharp \mathbb{m}_2^\sharp$ must be decidable
- The (binary) join $\mathbb{m}_1^\sharp \cup^\sharp \mathbb{m}_2^\sharp$ must be computable
- An iteration strategy ensuring termination (e.g., widening)

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$$

Abstract domain

- The abstract domain $\mathbb{M}^\sharp$ must be machine-representable
- The abstract test $\mathbb{m}_1^\sharp \subseteq^\sharp \mathbb{m}_2^\sharp$ must be decidable
- The (binary) join $\mathbb{m}_1^\sharp \cup^\sharp \mathbb{m}_2^\sharp$ must be computable
- An iteration strategy ensuring termination (e.g., widening)

Sound abstract transfer functions $(\!|\ {}^{\ell'} stmt\ |\!)^\sharp$ and $\mathbf{S}^\sharp_{stmt}$ on $\mathbb{M}^\sharp \to \mathbb{M}^\sharp$

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$$

Abstract domain

- The abstract domain $\mathbb{M}^\sharp$ must be machine-representable
- The abstract test $\mathfrak{m}_1^\sharp \subseteq^\sharp \mathfrak{m}_2^\sharp$ must be decidable
- The (binary) join $\mathfrak{m}_1^\sharp \cup^\sharp \mathfrak{m}_2^\sharp$ must be computable
- An iteration strategy ensuring termination (e.g., widening)

Sound abstract transfer functions $\llparenthesis\, {}^{\ell'} stmt \,\rrparenthesis^\sharp$ and $\mathbf{S}^\sharp_{stmt}$ on $\mathbb{M}^\sharp \rightarrow \mathbb{M}^\sharp$

Abstract transfer functions can be systematically derived from soundness proofs!

Define abstract transfer functions for conditions and code blocks, given $\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$

Conditions $(\!|\, {}^{\ell} bexp \,|\!)^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

$(\!|\, {}^{\ell} \mathtt{x} = exp \,|\!)^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ Code blocks

Define abstract transfer functions for conditions and code blocks, given $\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$

Conditions $\llparenthesis\, {}^{\ell} bexp \,\rrparenthesis^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

$\llparenthesis\, {}^{\ell} \mathtt{x} = exp \,\rrparenthesis^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ Code blocks

Soundness

$$\forall \mathfrak{m}^\sharp \in \mathbb{M}^\sharp$$

$$\llparenthesis\, {}^{\ell} bexp \,\rrparenthesis \circ \gamma(\mathfrak{m}^\sharp) \subseteq \gamma \circ \llparenthesis\, {}^{\ell} bexp \,\rrparenthesis^\sharp(\mathfrak{m}^\sharp)$$

$$\llparenthesis\, {}^{\ell} \mathtt{x} = exp \,\rrparenthesis \circ \gamma(\mathfrak{m}^\sharp) \subseteq \gamma \circ \llparenthesis\, {}^{\ell} \mathtt{x} = exp \,\rrparenthesis^\sharp(\mathfrak{m}^\sharp)$$

Define abstract transfer functions for conditions and code blocks, given $\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$

Conditions $\quad (\!|\, {}^{\ell} bexp \,|\!)^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

$(\!|\, {}^{\ell} \mathtt{x} = exp \,|\!)^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp \quad$ Code blocks

Soundness

$$\forall X \in \wp(\mathbb{M})$$

$$\alpha \circ (\!|\, {}^{\ell} bexp \,|\!) X \subseteq^\sharp (\!|\, {}^{\ell} bexp \,|\!)^\sharp \circ \alpha(X)$$

$$\alpha \circ (\!|\, {}^{\ell} \mathtt{x} = exp \,|\!) X \subseteq^\sharp (\!|\, {}^{\ell} \mathtt{x} = exp \,|\!)^\sharp \circ \alpha(X)$$

The abstract reachability semantics is the least solution of the system of equations

$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{m}^{\sharp}_{\top} \\ X_{\ell} \triangleq \bigcup^{\sharp}_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt \,|\!)^{\sharp} X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

The abstract reachability semantics is the least solution of the system of equations

$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{m}^\sharp_\top \\ X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis^\sharp X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

Each $X_\ell \mapsto \bigcup^\sharp_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis^\sharp X_{\ell'}$ is monotonic on the complete lattice $\langle \mathbb{M}^\sharp, \subseteq^\sharp, \cup^\sharp, \cap^\sharp, \mathbb{m}^\sharp_\bot, \mathbb{m}^\sharp_\top \rangle$

The abstract reachability semantics is the least solution of the system of equations

$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{m}^{\sharp}_{\top} \\ X_{\ell} \triangleq \bigcup^{\sharp}_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'} stmt \,\rrparenthesis^{\sharp} X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

Each $X_{\ell} \mapsto \bigcup^{\sharp}_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'} stmt \,\rrparenthesis^{\sharp} X_{\ell'}$ is monotonic on the complete lattice $\langle \mathbb{M}^{\sharp}, \subseteq^{\sharp}, \cup^{\sharp}, \cap^{\sharp}, \mathbb{m}^{\sharp}_{\bot}, \mathbb{m}^{\sharp}_{\top} \rangle$

- The least fixpoint does exist

The abstract reachability semantics is the least solution of the system of equations

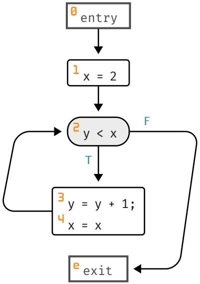
$$\begin{cases} X_{\mathtt{0}} \triangleq \mathbb{m}^\sharp_\top \\ X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt |\!)^\sharp X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

Each $X_\ell \mapsto \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt |\!)^\sharp X_{\ell'}$ is monotonic on the complete lattice $\langle \mathbb{M}^\sharp, \subseteq^\sharp, \cup^\sharp, \cap^\sharp, \mathbb{m}^\sharp_\bot, \mathbb{m}^\sharp_\top \rangle$

- The least fixpoint does exist
- The solution can be computed by increasing iterations

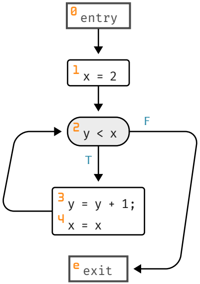
$$\begin{cases} X^0_{\mathtt{0}} \triangleq \mathbb{m}^\sharp_\top \\ X^0_\ell \triangleq \mathbb{m}^\sharp_\bot \end{cases} \qquad \begin{cases} X^{n+1}_{\mathtt{0}} \triangleq \mathbb{m}^\sharp_\top \\ X^{n+1}_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt |\!)^\sharp X^n_{\ell'} \end{cases}$$

Equational abstract reachability semantics: example



$$\begin{cases} X_{0} \triangleq \mathbb{m}^{\sharp}_{\top} \\ X_{\ell} \triangleq \bigcup^{\sharp}_{\ell' \in \mathbb{L}} (\!| \, {}^{\ell'}stmt \, |\!)^{\sharp} X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

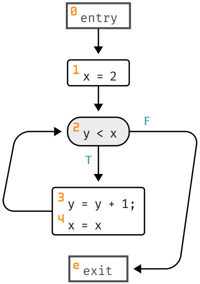
Equational abstract reachability semantics: example



$$\begin{cases} X_{0} \triangleq \mathfrak{m}^{\sharp}_{\top} \\ X_{\ell} \triangleq \bigcup^{\sharp}_{\ell' \in \mathbb{L}} (\!|\, {}^{\ell'}stmt \,|\!)^{\sharp} X_{\ell'} & \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

$$\begin{aligned}
X_{0} &= \mathfrak{m}^{\sharp}_{\top} \\
X_{1} &= (\!|\, {}^{0}\texttt{entry} \,|\!)^{\sharp} X_{0} \\
X_{2} &= (\!|\, {}^{1}\texttt{x = 2} \,|\!)^{\sharp} X_{1} \cup^{\sharp} (\!|\, {}^{4}\texttt{x = x} \,|\!)^{\sharp} X_{4} \\
X_{3} &= (\!|\, {}^{2}\texttt{y < x} \,|\!)^{\sharp} X_{2} \\
X_{4} &= (\!|\, {}^{3}\texttt{y = y + 1} \,|\!)^{\sharp} X_{3} \\
X_{e} &= (\!|\, {}^{2}\texttt{y >= x} \,|\!)^{\sharp} X_{2}
\end{aligned}$$

Equational abstract reachability semantics: example

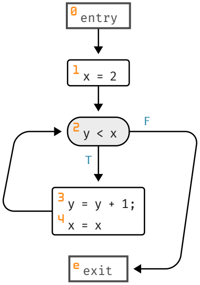


$$
\begin{cases}
X_0 \triangleq \mathfrak{m}^\sharp_\top \\
X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} \llparenthesis {}^{\ell'}stmt \rrparenthesis^\sharp X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell
\end{cases}
$$

$$
\begin{aligned}
X_0 &= \mathfrak{m}^\sharp_\top \\
X_1 &= \llparenthesis {}^{0}\texttt{entry} \rrparenthesis^\sharp X_0 \\
X_2 &= \llparenthesis {}^{1}\texttt{x = 2} \rrparenthesis^\sharp X_1 \cup^\sharp \llparenthesis {}^{4}\texttt{x = x} \rrparenthesis^\sharp X_4 \\
X_3 &= \llparenthesis {}^{2}\texttt{y < x} \rrparenthesis^\sharp X_2 \\
X_4 &= \llparenthesis {}^{3}\texttt{y = y + 1} \rrparenthesis^\sharp X_3 \\
X_e &= \llparenthesis {}^{2}\texttt{y >= x} \rrparenthesis^\sharp X_2
\end{aligned}
$$

X_0	X_1	X_2	X_3	X_4	X_e

Equational abstract reachability semantics: example

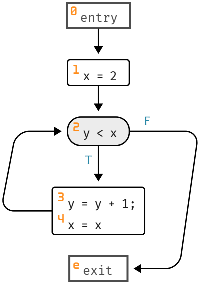


$$\begin{cases} X_0 \triangleq \mathfrak{m}^\sharp_\top \\ X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!|\, {}^{\ell'}stmt \,|\!)^\sharp X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\top^\pm, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$

$$X_0 = \mathfrak{m}^\sharp_\top$$
$$X_1 = (\!|\, {}^{0}\texttt{entry} \,|\!)^\sharp X_0$$
$$X_2 = (\!|\, {}^{1}\texttt{x = 2} \,|\!)^\sharp X_1 \cup^\sharp (\!|\, {}^{4}\texttt{x = x} \,|\!)^\sharp X_4$$
$$X_3 = (\!|\, {}^{2}\texttt{y < x} \,|\!)^\sharp X_2$$
$$X_4 = (\!|\, {}^{3}\texttt{y = y + 1} \,|\!)^\sharp X_3$$
$$X_e = (\!|\, {}^{2}\texttt{y >= x} \,|\!)^\sharp X_2$$

Equational abstract reachability semantics: example

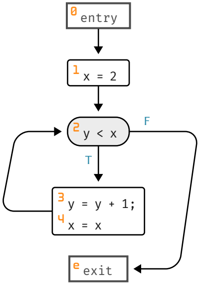


$$\begin{cases} X_{0} \triangleq \mathfrak{m}^{\sharp}_{\top} \\ X_{\ell} \triangleq \bigcup^{\sharp}_{\ell' \in \mathbb{L}} \llparenthesis\, {}^{\ell'}stmt \,\rrparenthesis^{\sharp} X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\top^{\pm}, \top^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$
$(\top^{\pm}, \top^{\pm})$	$(\top^{\pm}, \top^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$

$$\begin{aligned}
X_{0} &= \mathfrak{m}^{\sharp}_{\top} \\
X_{1} &= \llparenthesis\, {}^{0}\texttt{entry} \,\rrparenthesis^{\sharp} X_{0} \\
X_{2} &= \llparenthesis\, {}^{1}\texttt{x = 2} \,\rrparenthesis^{\sharp} X_{1} \cup^{\sharp} \llparenthesis\, {}^{4}\texttt{x = x} \,\rrparenthesis^{\sharp} X_{4} \\
X_{3} &= \llparenthesis\, {}^{2}\texttt{y < x} \,\rrparenthesis^{\sharp} X_{2} \\
X_{4} &= \llparenthesis\, {}^{3}\texttt{y = y + 1} \,\rrparenthesis^{\sharp} X_{3} \\
X_{e} &= \llparenthesis\, {}^{2}\texttt{y >= x} \,\rrparenthesis^{\sharp} X_{2}
\end{aligned}$$

Equational abstract reachability semantics: example

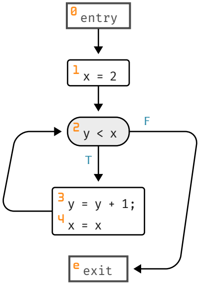


$$\begin{cases} X_0 \triangleq \mathfrak{m}^\sharp_\top \\ X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt |\!)^\sharp X_{\ell'} \quad \text{if next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

$$X_0 = \mathfrak{m}^\sharp_\top$$
$$X_1 = (\!| {}^{0}\texttt{entry} |\!)^\sharp X_0$$
$$X_2 = (\!| {}^{1}\texttt{x = 2} |\!)^\sharp X_1 \cup^\sharp (\!| {}^{4}\texttt{x = x} |\!)^\sharp X_4$$
$$X_3 = (\!| {}^{2}\texttt{y < x} |\!)^\sharp X_2$$
$$X_4 = (\!| {}^{3}\texttt{y = y + 1} |\!)^\sharp X_3$$
$$X_e = (\!| {}^{2}\texttt{y >= x} |\!)^\sharp X_2$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\top^\pm, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(+0, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$

Equational abstract reachability semantics: example

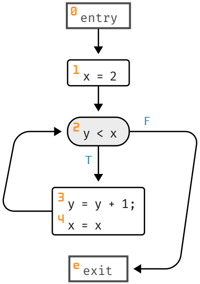


$$\begin{cases} X_0 \triangleq \mathfrak{m}^\sharp_\top \\ X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!)^\sharp X_{\ell'} \quad \text{if } \mathrm{next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

$$\begin{aligned} X_0 &= \mathfrak{m}^\sharp_\top \\ X_1 &= (\!| {}^{0}\texttt{entry} |\!)^\sharp X_0 \\ X_2 &= (\!| {}^{1}\texttt{x = 2} |\!)^\sharp X_1 \cup^\sharp (\!| {}^{4}\texttt{x = x} |\!)^\sharp X_4 \\ X_3 &= (\!| {}^{2}\texttt{y < x} |\!)^\sharp X_2 \\ X_4 &= (\!| {}^{3}\texttt{y = y + 1} |\!)^\sharp X_3 \\ X_e &= (\!| {}^{2}\texttt{y >= x} |\!)^\sharp X_2 \end{aligned}$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\top^\pm, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(+0, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(+0, \top^\pm)$	$(+0, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$

Equational abstract reachability semantics: example

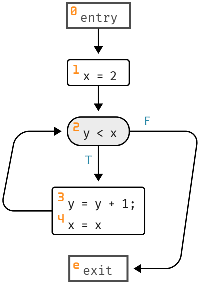


$$\begin{cases} X_{0} \triangleq \mathfrak{m}^{\sharp}_{\top} \\ X_{\ell} \triangleq \bigcup^{\sharp}_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!)^{\sharp} X_{\ell'} \quad \text{if next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

$$\begin{aligned}
X_{0} &= \mathfrak{m}^{\sharp}_{\top} \\
X_{1} &= (\!| {}^{0}\texttt{entry} |\!)^{\sharp} X_{0} \\
X_{2} &= (\!| {}^{1}\texttt{x = 2} |\!)^{\sharp} X_{1} \cup^{\sharp} (\!| {}^{4}\texttt{x = x} |\!)^{\sharp} X_{4} \\
X_{3} &= (\!| {}^{2}\texttt{y < x} |\!)^{\sharp} X_{2} \\
X_{4} &= (\!| {}^{3}\texttt{y = y + 1} |\!)^{\sharp} X_{3} \\
X_{e} &= (\!| {}^{2}\texttt{y >= x} |\!)^{\sharp} X_{2}
\end{aligned}$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\top^{\pm}, \top^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$
$(\top^{\pm}, \top^{\pm})$	$(\top^{\pm}, \top^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$
$(\top^{\pm}, \top^{\pm})$	$(\top^{\pm}, \top^{\pm})$	$(+0, \top^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$
$(\top^{\pm}, \top^{\pm})$	$(\top^{\pm}, \top^{\pm})$	$(+0, \top^{\pm})$	$(+0, \top^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$
$(\top^{\pm}, \top^{\pm})$	$(\top^{\pm}, \top^{\pm})$	$(+0, \top^{\pm})$	$(+0, \top^{\pm})$	$(+0, \top^{\pm})$	$(\bot^{\pm}, \bot^{\pm})$

Equational abstract reachability semantics: example



$$\begin{cases} X_0 \triangleq \mathfrak{m}^\sharp_\top \\ X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt |\!)^\sharp X_{\ell'} \quad \text{if } \mathrm{next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

$$X_0 = \mathfrak{m}^\sharp_\top$$
$$X_1 = (\!| {}^{0}\texttt{entry} |\!)^\sharp X_0$$
$$X_2 = (\!| {}^{1}\texttt{x = 2} |\!)^\sharp X_1 \cup^\sharp (\!| {}^{4}\texttt{x = x} |\!)^\sharp X_4$$
$$X_3 = (\!| {}^{2}\texttt{y < x} |\!)^\sharp X_2$$
$$X_4 = (\!| {}^{3}\texttt{y = y + 1} |\!)^\sharp X_3$$
$$X_e = (\!| {}^{2}\texttt{y >= x} |\!)^\sharp X_2$$

X_0	X_1	X_2	X_3	X_4	X_e
$(\top^\pm, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(+0, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(+0, \top^\pm)$	$(+0, \top^\pm)$	$(\bot^\pm, \bot^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(+0, \top^\pm)$	$(+0, \top^\pm)$	$(+0, \top^\pm)$	$(\bot^\pm, \bot^\pm)$
$(\top^\pm, \top^\pm)$	$(\top^\pm, \top^\pm)$	$(+0, \top^\pm)$	$(+0, \top^\pm)$	$(+0, \top^\pm)$	$(+0, +0)$

Abstract post-condition semantics $\llbracket P \rrbracket^\sharp$, given $\langle \wp(\mathbb{M}), \subseteq \rangle \xleftrightarrow[\alpha]{\gamma} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$

Abstract post-condition semantics $[\![P]\!]^\sharp$, given $\langle \wp(\mathbb{M}), \subseteq\rangle \xleftarrow[\alpha]{\gamma} \langle \mathbb{M}^\sharp, \subseteq^\sharp\rangle$

- Inductively computed on program syntax $[\![P]\!]^\sharp \triangleq \mathbf{S}^\sharp_{stmt_n} \circ \ldots \circ \mathbf{S}^\sharp_{stmt_1}(\mathfrak{I}^\sharp)$ // $\mathfrak{I}^\sharp \triangleq \alpha(\mathfrak{I} \subseteq \mathbb{M})$
- Given an abstract transfer function $\mathbf{S}^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ for all statements $stmt_1, \ldots, stmt_n$ of P

Abstract post-condition semantics $[\![P]\!]^\sharp$, given $\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$

- Inductively computed on program syntax $[\![P]\!]^\sharp \triangleq \mathbf{S}^\sharp_{stmt_n} \circ \ldots \circ \mathbf{S}^\sharp_{stmt_1}(\mathfrak{I}^\sharp)$ // $\mathfrak{I}^\sharp \triangleq \alpha(\mathfrak{I} \subseteq \mathbb{M})$
- Given an abstract transfer function $\mathbf{S}^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ for all statements $stmt_1, \ldots, stmt_n$ of P

Boolean expressions $\mathbf{S}^\sharp_{bexp} : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

$\mathbf{S}^\sharp_{\mathtt{x}=exp} : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ Assignments

Abstract post-condition semantics $[\![P]\!]^\sharp$, given $\langle \wp(\mathbb{M}), \subseteq \rangle \xleftrightarrow[\alpha]{\gamma} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$

- Inductively computed on program syntax $[\![P]\!]^\sharp \triangleq \mathbf{S}^\sharp_{stmt_n} \circ \ldots \circ \mathbf{S}^\sharp_{stmt_1}(\mathfrak{I}^\sharp)$ // $\mathfrak{I}^\sharp \triangleq \alpha(\mathfrak{I} \subseteq \mathbb{M})$
- Given an abstract transfer function $\mathbf{S}^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ for all statements $stmt_1, \ldots, stmt_n$ of P

Boolean expressions $\mathbf{S}^\sharp_{bexp} : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

$\mathbf{S}^\sharp_{\mathtt{x}=exp} : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ Assignments

Soundness

$$\forall \mathfrak{m}^\sharp \in \mathbb{M}^\sharp$$

$$\mathbf{S}^c_{bexp} \circ \gamma(\mathfrak{m}^\sharp) \subseteq \gamma \circ \mathbf{S}^\sharp_{bexp}(\mathfrak{m}^\sharp)$$

$$\mathbf{S}^c_{bexp} \circ \gamma(\mathfrak{m}^\sharp) \subseteq \gamma \circ \mathbf{S}^\sharp_{bexp}$$

Abstract post-condition semantics $[\![P]\!]^\sharp$, given $\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$

- Inductively computed on program syntax $[\![P]\!]^\sharp \triangleq \mathbf{S}^\sharp_{stmt_n} \circ \ldots \circ \mathbf{S}^\sharp_{stmt_1}(\mathfrak{I}^\sharp)$ // $\mathfrak{I}^\sharp \triangleq \alpha(\mathfrak{I} \subseteq \mathbb{M})$
- Given an abstract transfer function $\mathbf{S}^\sharp : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ for all statements $stmt_1, \ldots, stmt_n$ of P

Boolean expressions $\mathbf{S}^\sharp_{bexp} : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

$\mathbf{S}^\sharp_{\mathtt{X}=exp} : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ Assignments

Soundness

$$\forall X \in \wp(\mathbb{M})$$

$$\alpha \circ \mathbf{S}^c_{\mathtt{X}=exp}(X) \subseteq^\sharp \mathbf{S}^\sharp_{\mathtt{X}=exp} \circ \alpha(X)$$

$$\alpha \circ \mathbf{S}^c_{\mathtt{X}=exp}(X) \subseteq^\sharp \mathbf{S}^\sharp_{\mathtt{X}=exp} \circ \alpha(X)$$

Programs $\mathbf{S}^{\sharp}_{P} : \mathbb{M}^{\sharp} \rightarrow \mathbb{M}^{\sharp}$

Inductive cases

Abstract Post-condition semantics (Cont'd)

Programs $\mathbf{S}^\sharp_P : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

Inductive cases

- $\mathbf{S}^\sharp_{stmt_1\,;\,stmt_2}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{stmt_2} \circ \mathbf{S}^\sharp_{stmt_1}(\mathfrak{m}^\sharp)$
- $\mathbf{S}^\sharp_{\mathtt{if}(bexp)\{stmt_1\}\mathtt{else}\{stmt_2\}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{stmt_1} \circ \mathbf{S}^\sharp_{bexp}(\mathfrak{m}^\sharp) \cup^\sharp \mathbf{S}^\sharp_{stmt_2} \circ \mathbf{S}^\sharp_{\neg bexp}(\mathfrak{m}^\sharp)$

Programs $\mathbf{S}^\sharp_P : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

Inductive cases

- $\mathbf{S}^\sharp_{stmt_1\,;\,stmt_2}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{stmt_2} \circ \mathbf{S}^\sharp_{stmt_1}(\mathfrak{m}^\sharp)$
- $\mathbf{S}^\sharp_{\texttt{if}(bexp)\{stmt_1\}\texttt{else}\{stmt_2\}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{stmt_1} \circ \mathbf{S}^\sharp_{bexp}(\mathfrak{m}^\sharp) \cup^\sharp \mathbf{S}^\sharp_{stmt_2} \circ \mathbf{S}^\sharp_{\neg bexp}(\mathfrak{m}^\sharp)$
- $\mathbf{S}^\sharp_{\texttt{while}(bexp)\{stmt\}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\neg bexp} \circ \left(\mathsf{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathfrak{m}}^\sharp .\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{stmt} \circ \mathbf{S}^c_{bexp}(\overline{\mathfrak{m}}^\sharp)\right)$

Programs $\mathbf{S}^\sharp_P : \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

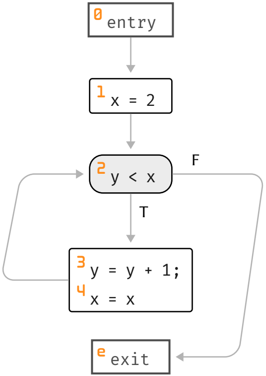
Inductive cases

- $\mathbf{S}^\sharp_{stmt_1\,;\,stmt_2}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{stmt_2} \circ \mathbf{S}^\sharp_{stmt_1}(\mathfrak{m}^\sharp)$
- $\mathbf{S}^\sharp_{\texttt{if}(bexp)\{stmt_1\}\texttt{else}\{stmt_2\}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{stmt_1} \circ \mathbf{S}^\sharp_{bexp}(\mathfrak{m}^\sharp) \cup^\sharp \mathbf{S}^\sharp_{stmt_2} \circ \mathbf{S}^\sharp_{\neg bexp}(\mathfrak{m}^\sharp)$
- $\mathbf{S}^\sharp_{\texttt{while}(bexp)\{stmt\}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\neg bexp} \circ (\mathsf{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathfrak{m}}^\sharp\,.\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{stmt} \circ \mathbf{S}^{c}_{bexp}(\overline{\mathfrak{m}}^\sharp))$

Soundness

$$[\![P]\!]^c \subseteq \gamma([\![P]\!]^\sharp) \qquad\qquad \alpha([\![P]\!]^c) \subseteq^\sharp [\![P]\!]^\sharp$$

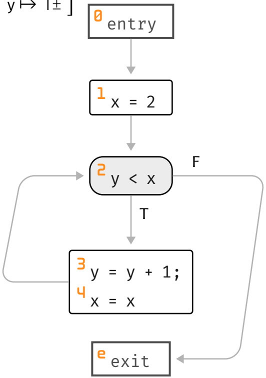
Abstract Post-condition semantics: example



Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\pm$

Abstract Post-condition semantics: example

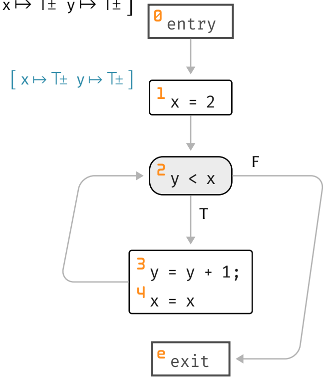
Input: $[\, x \mapsto \top^{\pm} \;\; y \mapsto \top^{\pm} \,]$



Abstract memories: $\mathbb{M}^{\sharp} = \mathbb{X} \to \mathbb{Z}^{\pm}$

Abstract Post-condition semantics: example

Input: $[\, x \mapsto \top_{\pm} \;\; y \mapsto \top_{\pm} \,]$

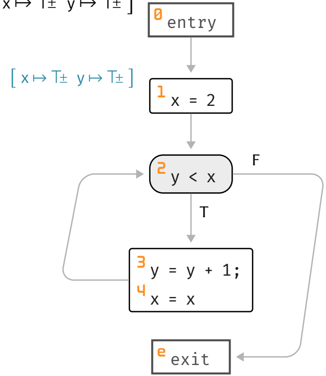


Abstract memories: $\mathbb{M}^{\sharp} = \mathbb{X} \to \mathbb{Z}^{\pm}$

Abstract Post-condition semantics: example

$$\mathbf{S}^{\sharp}_{\mathtt{x=2}}(\mathfrak{m}^{\sharp}) \triangleq \mathfrak{m}^{\sharp}[\mathtt{x} \leftarrow \alpha(2)]$$

Input: $[\, \mathtt{x} \mapsto \top_{\pm} \;\; \mathtt{y} \mapsto \top_{\pm} \,]$

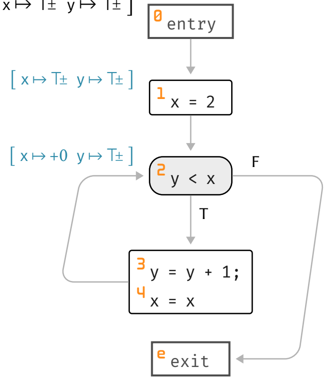


Abstract memories: $\mathbb{M}^{\sharp} = \mathbb{X} \to \mathbb{Z}^{\pm}$

Abstract Post-condition semantics: example

$$\mathbf{S}^{\sharp}_{\mathtt{x=2}}(\mathbb{m}^{\sharp}) \triangleq \mathbb{m}^{\sharp}[\mathtt{x} \leftarrow \alpha(2)] = [\, \mathtt{x} \mapsto +0 \;\; \mathtt{y} \mapsto \top\pm \,]$$

Input: $[\, \mathtt{x} \mapsto \top\pm \;\; \mathtt{y} \mapsto \top\pm \,]$

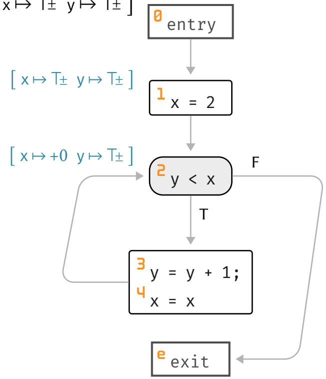


Abstract memories: $\mathbb{M}^{\sharp} = \mathbb{X} \to \mathbb{Z}^{\pm}$

Abstract Post-condition semantics: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathbb{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}\left(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathbb{m}}^\sharp\ .\ \mathbb{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathbb{m}}^\sharp)\right)$$

Input: $[\ x \mapsto \top\pm\ \ y \mapsto \top\pm\ ]$

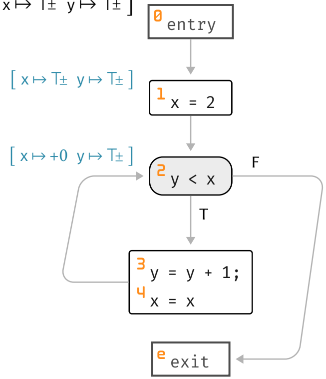


Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\pm$

Abstract Post-condition semantics: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}(\mathsf{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathfrak{m}}^\sharp\ .\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathfrak{m}}^\sharp))$$

Input: $[\ x \mapsto \top_\pm\ \ y \mapsto \top_\pm\ ]$



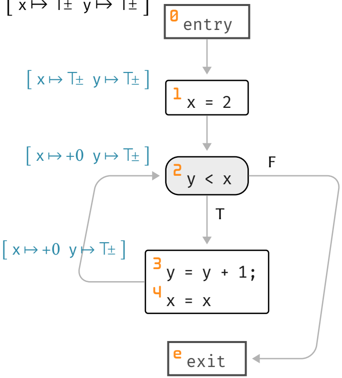
$X^0 = [\ x \mapsto +0\ \ y \mapsto \top_\pm\ ]$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\pm$

Abstract Post-condition semantics: example

$$\mathbf{S}^{\sharp}_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^{\sharp}) \triangleq \mathbf{S}^{\sharp}_{\texttt{x>=y}}\left(\mathrm{lfp}^{\subseteq^{\sharp}}\ \lambda\overline{\mathfrak{m}}^{\sharp}.\ \mathfrak{m}^{\sharp} \cup^{\sharp} \mathbf{S}^{\sharp}_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^{\sharp}_{\texttt{x<y}}(\overline{\mathfrak{m}}^{\sharp})\right)$$

Input: $[\,\texttt{x} \mapsto \top^{\pm}\ \ \texttt{y} \mapsto \top^{\pm}\,]$



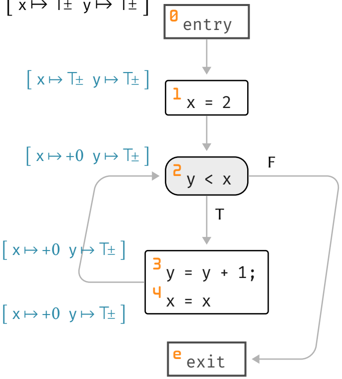
$$X^0 = [\,\texttt{x} \mapsto +0\ \ \texttt{y} \mapsto \top^{\pm}\,]$$

Abstract memories: $\mathbb{M}^{\sharp} = \mathbb{X} \to \mathbb{Z}^{\pm}$

Abstract Post-condition semantics: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathbb{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathbb{m}}^\sharp\ .\ \mathbb{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathbb{m}}^\sharp))$$

Input: $[\,x \mapsto \top_\pm\ \ y \mapsto \top_\pm\,]$



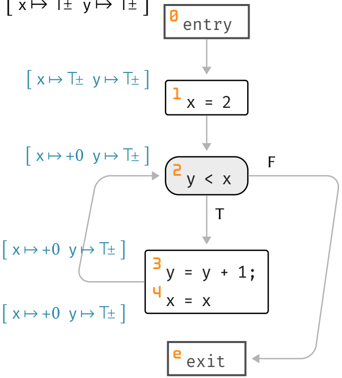
$X^0 = [\,x \mapsto +0\ \ y \mapsto \top_\pm\,]$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\pm$

Abstract Post-condition semantics: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}\Big(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda \overline{\mathfrak{m}}^\sharp\,.\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathfrak{m}}^\sharp)\Big)$$

Input: $[\,\mathtt{x} \mapsto \top_\pm \ \ \mathtt{y} \mapsto \top_\pm\,]$



$$X^0 = [\,\mathtt{x} \mapsto +0 \ \ \mathtt{y} \mapsto \top_\pm\,]$$

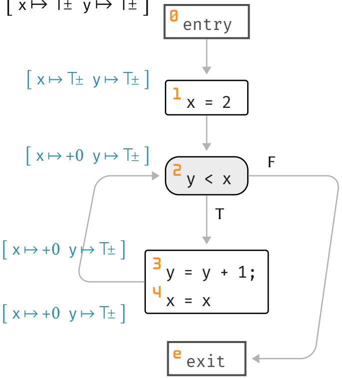
$$X^1 = [\,\mathtt{x} \mapsto +0 \ \ \mathtt{y} \mapsto \top_\pm\,]$$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\pm$

Abstract Post-condition semantics: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathbb{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda \overline{\mathbb{m}}^\sharp .\ \mathbb{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathbb{m}}^\sharp))$$

Input: $[\, x \mapsto \top_\pm \ \ y \mapsto \top_\pm \,]$



$X^0 = [\, x \mapsto +0 \ \ y \mapsto \top_\pm \,]$

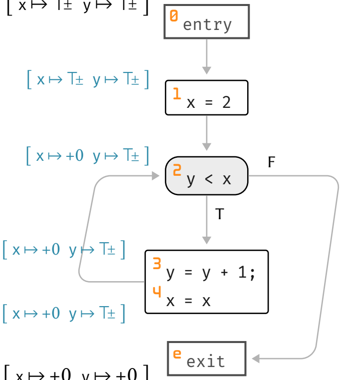
$\mathrm{lfp} = [\, x \mapsto +0 \ \ y \mapsto \top_\pm \,]$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\pm$

Abstract Post-condition semantics: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathbb{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}\left(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda \overline{\mathbb{m}}^\sharp\,.\ \mathbb{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathbb{m}}^\sharp)\right)$$

Input: $[\,x \mapsto \top_\pm \;\; y \mapsto \top_\pm\,]$



Output: $[\,x \mapsto +0 \;\; y \mapsto +0\,]$

$X^0 = [\,x \mapsto +0 \;\; y \mapsto \top_\pm\,]$

$\mathrm{lfp} = [\,x \mapsto +0 \;\; y \mapsto \top_\pm\,]$

Output: $[\,x \mapsto +0 \;\; y \mapsto +0\,]$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\pm$

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

Non-relational Abstraction

Forget all relations between variables

$$\langle \wp(\mathbb{X} \to \mathbb{V}), \subseteq \rangle \underset{\alpha_{nr}}{\overset{\gamma_{nr}}{\leftrightarrows}} \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle$$

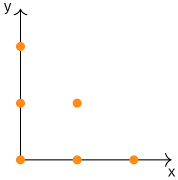
where
$$\alpha_{nr} \triangleq \lambda X \, . \, \lambda \mathbb{x} \, . \, \{ \mathbb{m}(\mathbb{x}) \in \mathbb{V} \mid \mathbb{m} \in X \} \qquad \gamma_{nr} \triangleq \lambda \overline{\mathbb{m}} \, . \, \{ \mathbb{m} \in \mathbb{M} \mid \forall \mathbb{x} \in \mathbb{X} \, . \, \mathbb{m}(\mathbb{x}) \in \overline{\mathbb{m}}(\mathbb{x}) \}$$

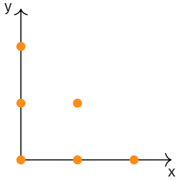
UNIVERSITÀ di VERONA Dipartimento di INFORMATICA

Forget all relations between variables

$$\langle \wp(\mathbb{X} \to \mathbb{V}), \subseteq \rangle \underset{\alpha_{nr}}{\overset{\gamma_{nr}}{\leftrightarrows}} \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle$$

where
$\alpha_{nr} \triangleq \lambda X \,.\, \lambda \mathbb{x} \,.\, \{\mathbb{m}(\mathbb{x}) \in \mathbb{V} \mid \mathbb{m} \in X\}$ $\gamma_{nr} \triangleq \lambda \overline{\mathbb{m}} \,.\, \{\mathbb{m} \in \mathbb{M} \mid \forall \mathbb{x} \in \mathbb{X} \,.\, \mathbb{m}(\mathbb{x}) \in \overline{\mathbb{m}}(\mathbb{x})\}$



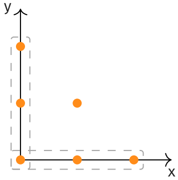


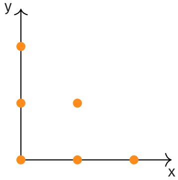
$\{\mathbb{m} \in \mathbb{M} \mid \mathbb{m}(\mathrm{x}), \mathbb{m}(\mathrm{y}) \in [0, 2] \wedge \mathrm{x} + \mathrm{y} \leq 2\}$

Forget all relations between variables

$$\langle \wp(\mathbb{X} \to \mathbb{V}), \subseteq \rangle \underset{\alpha_{nr}}{\overset{\gamma_{nr}}{\leftrightarrows}} \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle$$

where $\alpha_{nr} \triangleq \lambda X.\ \lambda \mathbb{x}.\ \{\mathbb{m}(\mathbb{x}) \in \mathbb{V} \mid \mathbb{m} \in X\}$ $\qquad$ $\gamma_{nr} \triangleq \lambda \overline{\mathbb{m}}.\ \{\mathbb{m} \in \mathbb{M} \mid \forall \mathbb{x} \in \mathbb{X}.\ \mathbb{m}(\mathbb{x}) \in \overline{\mathbb{m}}(\mathbb{x})\}$



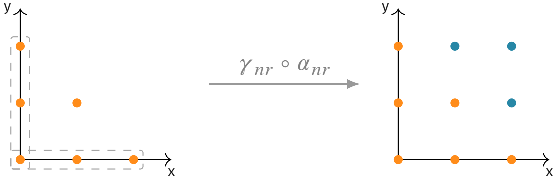


$\{\mathbb{m} \in \mathbb{M} \mid \mathbb{m}(\mathsf{x}), \mathbb{m}(\mathsf{y}) \in [0, 2] \wedge \mathsf{x} + \mathsf{y} \leq 2\}$

Forget all relations between variables

$$\langle \wp(\mathbb{X} \to \mathbb{V}), \subseteq \rangle \underset{\alpha_{nr}}{\overset{\gamma_{nr}}{\leftrightarrows}} \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle$$

where $\alpha_{nr} \triangleq \lambda X \,.\, \lambda \mathbb{x} \,.\, \{\mathbb{m}(\mathbb{x}) \in \mathbb{V} \mid \mathbb{m} \in X\}$ $\quad \gamma_{nr} \triangleq \lambda \overline{\mathbb{m}} \,.\, \{\mathbb{m} \in \mathbb{M} \mid \forall \mathbb{x} \in \mathbb{X} \,.\, \mathbb{m}(\mathbb{x}) \in \overline{\mathbb{m}}(\mathbb{x})\}$



$\{\mathbb{m} \in \mathbb{M} \mid \mathbb{m}(\mathsf{x}), \mathbb{m}(\mathsf{y}) \in [0, 2] \land \mathsf{x} + \mathsf{y} \leq 2\}$ $\quad \{\mathbb{m} \in \mathbb{M} \mid \mathbb{m}(\mathsf{x}) \in [0, 2] \land \mathbb{m}(\mathsf{y}) \in [0, 2]\}$

Abstraction $\mathbb{V}^\sharp$ of values $\wp(\mathbb{V})$

on single variables

Abstraction $\mathbb{V}^\sharp$ of values $\wp(\mathbb{V})$ — on single variables

Ingredients

$\mathbb{V}^\sharp$	machine-representable abstract values
$\alpha^v : \wp(\mathbb{V}) \to \mathbb{V}^\sharp$	abstraction (encoding)
$\gamma^v : \mathbb{V}^\sharp \to \wp(\mathbb{V})$	concretization (decoding)
$\subseteq_v$	computable abstract partial order
$\perp^v$ and $\top^v$	abstract representation of $\emptyset$ and $\mathbb{V}$
$\cup^v$ and $\cap^v$	abstract version of $\cup$ and $\cap$
∇^v and Δ^v	extrapolation and interpolation operators

Define a sound version of arithmetic operations in $\mathbb{V}^\sharp$

$$
\begin{aligned}
\{\text{-}\mathbb{v} \mid \mathbb{v} \in \gamma^{v}(\mathbb{v}^\sharp)\} &\subseteq \gamma^{v}(\text{-}_{v}\mathbb{v}^\sharp) \\
\{\mathbb{v}_1 + \mathbb{v}_2 \mid \mathbb{v}_1 \in \gamma^{v}(\mathbb{v}_1^\sharp) \wedge \mathbb{v}_2 \in \gamma^{v}(\mathbb{v}_2^\sharp)\} &\subseteq \gamma^{v}(\mathbb{v}_1^\sharp +_{v} \mathbb{v}_2^\sharp) \\
&\dots
\end{aligned}
$$

Define a sound version of arithmetic operations in $\mathbb{V}^\sharp$

$$\begin{aligned}\{\text{-}\mathbb{v} \mid \mathbb{v} \in \gamma^{v}(\mathbb{v}^\sharp)\} &\subseteq \gamma^{v}(\text{-}_v \mathbb{v}^\sharp)\\ \{\mathbb{v}_1 + \mathbb{v}_2 \mid \mathbb{v}_1 \in \gamma^{v}(\mathbb{v}_1^\sharp) \wedge \mathbb{v}_2 \in \gamma^{v}(\mathbb{v}_2^\sharp)\} &\subseteq \gamma^{v}(\mathbb{v}_1^\sharp +_v \mathbb{v}_2^\sharp)\\ &\dots\end{aligned}$$

Approximate the best correct abstract operations in the case of Galois Connections

$$\begin{aligned}\text{-}_v \mathbb{v}^\sharp &\triangleq \alpha_v(\{\text{-}\mathbb{v} \mid \mathbb{v} \in \gamma^{v}(\mathbb{v}^\sharp)\})\\ \mathbb{v}_1^\sharp +_v \mathbb{v}_2^\sharp &\triangleq \alpha_v(\{\mathbb{v}_1 + \mathbb{v}_2 \mid \mathbb{v}_1 \in \gamma^{v}(\mathbb{v}_1^\sharp) \wedge \mathbb{v}_2 \in \gamma^{v}(\mathbb{v}_2^\sharp)\})\\ &\dots\end{aligned}$$

Lift to non-relational program memories

$$\langle \wp(\mathbb{V}), \subseteq \rangle \underset{\alpha^{\nu}}{\overset{\gamma^{\nu}}{\leftrightarrows}} \langle \mathbb{V}^{\sharp}, \subseteq_{\nu} \rangle$$

Lift to non-relational program memories

$$\langle \wp(\mathbb{V}), \subseteq \rangle \underset{\alpha^{v}}{\overset{\gamma^{v}}{\leftrightarrows}} \langle \mathbb{V}^{\sharp}, \subseteq_{v} \rangle \quad \text{lift to set } \mathbb{X} \quad \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle \underset{\lambda f\,.\,\alpha^{v} \circ f}{\overset{\lambda f\,.\,\gamma^{v} \circ f}{\leftrightarrows}} \langle \mathbb{X} \to \mathbb{V}^{\sharp}, \dot{\subseteq}^{\sharp}_{v} \rangle$$

Lift to non-relational program memories

Associate an abstract value to each variable

$$\langle \wp(\mathbb{V}), \subseteq \rangle \underset{\alpha^{v}}{\overset{\gamma^{v}}{\leftrightarrows}} \langle \mathbb{V}^{\sharp}, \subseteq_{v} \rangle \quad \text{lift to set } \mathbb{X} \quad \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle \underset{\lambda \overline{\mathbb{m}} .\ \alpha^{v} \circ \overline{\mathbb{m}}}{\overset{\lambda \overline{\mathbb{m}} .\ \gamma^{v} \circ \overline{\mathbb{m}}}{\leftrightarrows}} \langle \mathbb{X} \to \mathbb{V}^{\sharp}, \dot{\subseteq}^{\sharp}_{v} \rangle$$

Associate an abstract value to each variable

$$\langle \wp(\mathbb{V}), \subseteq \rangle \underset{\alpha^{v}}{\overset{\gamma^{v}}{\leftrightarrows}} \langle \mathbb{V}^{\sharp}, \subseteq_{v} \rangle \quad \text{lift to set } \mathbb{X} \quad \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle \underset{\lambda \overline{\mathfrak{m}}.\ \alpha^{v} \circ \overline{\mathfrak{m}}}{\overset{\lambda \overline{\mathfrak{m}}.\ \gamma^{v} \circ \overline{\mathfrak{m}}}{\leftrightarrows}} \langle \mathbb{X} \to \mathbb{V}^{\sharp}, \dot{\subseteq}^{\sharp}_{v} \rangle$$

By composition we have with $\mathbb{M}^{\sharp} \triangleq \mathbb{X} \to \mathbb{V}^{\sharp}$ and $\subseteq^{\sharp} \triangleq \dot{\subseteq}_{v}$

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha^{v}_{nr}}{\overset{\gamma^{v}_{nr}}{\leftrightarrows}} \langle \mathbb{M}^{\sharp}, \subseteq^{\sharp} \rangle$$

Associate an abstract value to each variable

$$\langle \wp(\mathbb{V}), \subseteq \rangle \underset{\alpha^{v}}{\overset{\gamma^{v}}{\leftrightarrows}} \langle \mathbb{V}^{\sharp}, \subseteq_{v} \rangle \quad \text{lift to set } \mathbb{X} \quad \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle \underset{\lambda \overline{\mathrm{m}}\,.\, \alpha^{v} \circ \overline{\mathrm{m}}}{\overset{\lambda \overline{\mathrm{m}}\,.\, \gamma^{v} \circ \overline{\mathrm{m}}}{\leftrightarrows}} \langle \mathbb{X} \to \mathbb{V}^{\sharp}, \dot{\subseteq}^{\sharp}_{v} \rangle$$

By composition we have with $\mathbb{M}^{\sharp} \triangleq \mathbb{X} \to \mathbb{V}^{\sharp}$ and $\subseteq^{\sharp} \triangleq \dot{\subseteq}_{v}$

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha^{v}_{nr}}{\overset{\gamma^{v}_{nr}}{\leftrightarrows}} \langle \mathbb{M}^{\sharp}, \subseteq^{\sharp} \rangle$$

$\bot^{\sharp} \triangleq \lambda \mathrm{x}\,.\, \bot^{v}$ and $\top^{\sharp} \triangleq \lambda \mathrm{x}\,.\, \top^{v}$

Associate an abstract value to each variable

$$\langle \wp(\mathbb{V}), \subseteq \rangle \underset{\alpha^{v}}{\overset{\gamma^{v}}{\leftrightarrows}} \langle \mathbb{V}^{\sharp}, \subseteq_{v} \rangle \quad \text{lift to set } \mathbb{X} \quad \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle \underset{\lambda \overline{\mathbb{m}}\,.\, \alpha^{v} \circ \overline{\mathbb{m}}}{\overset{\lambda \overline{\mathbb{m}}\,.\, \gamma^{v} \circ \overline{\mathbb{m}}}{\leftrightarrows}} \langle \mathbb{X} \to \mathbb{V}^{\sharp}, \dot{\subseteq}^{\sharp}_{v} \rangle$$

By composition we have with $\mathbb{M}^{\sharp} \triangleq \mathbb{X} \to \mathbb{V}^{\sharp}$ and $\subseteq^{\sharp} \triangleq \dot{\subseteq}_{v}$

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha^{v}_{nr}}{\overset{\gamma^{v}_{nr}}{\leftrightarrows}} \langle \mathbb{M}^{\sharp}, \subseteq^{\sharp} \rangle$$

$\bot^{\sharp} \triangleq \lambda \mathrm{x}\,.\, \bot^{v}$ and $\top^{\sharp} \triangleq \lambda \mathrm{x}\,.\, \top^{v}$

$\alpha^{v}_{nr} \triangleq \lambda X\,.\, \llbracket X = \emptyset \,?\, \bot^{\sharp} \,:\, \lambda \mathrm{x}\,.\, \alpha^{v}(\{\mathbb{m}(\mathrm{x}) \in \mathbb{V} \mid \mathbb{m} \in X\}) \rrbracket$

Associate an abstract value to each variable

$$\langle \wp(\mathbb{V}), \subseteq \rangle \underset{\alpha^v}{\overset{\gamma^v}{\leftrightarrows}} \langle \mathbb{V}^\sharp, \subseteq_v \rangle \quad \text{lift to set } \mathbb{X} \quad \langle \mathbb{X} \to \wp(\mathbb{V}), \dot{\subseteq} \rangle \underset{\lambda \overline{\mathbb{m}}\,.\, \alpha^v \circ \overline{\mathbb{m}}}{\overset{\lambda \overline{\mathbb{m}}\,.\, \gamma^v \circ \overline{\mathbb{m}}}{\leftrightarrows}} \langle \mathbb{X} \to \mathbb{V}^\sharp, \dot{\subseteq}^\sharp_v \rangle$$

By composition we have with $\mathbb{M}^\sharp \triangleq \mathbb{X} \to \mathbb{V}^\sharp$ and $\subseteq^\sharp \triangleq \dot{\subseteq}_v$

$$\langle \wp(\mathbb{M}), \subseteq \rangle \underset{\alpha^v_{nr}}{\overset{\gamma^v_{nr}}{\leftrightarrows}} \langle \mathbb{M}^\sharp, \subseteq^\sharp \rangle$$

$\bot^\sharp \triangleq \lambda \mathsf{x}\,.\, \bot^v$ and $\top^\sharp \triangleq \lambda \mathsf{x}\,.\, \top^v$

$\alpha^v_{nr} \triangleq \lambda X\,.\, \llparenthesis X = \emptyset \ ?\ \bot^\sharp \ :\ \lambda \mathsf{x}\,.\, \alpha^v(\{\mathbb{m}(\mathsf{x}) \in \mathbb{V} \mid \mathbb{m} \in X\}) \rrparenthesis$

$\gamma^v_{nr} \triangleq \lambda \mathbb{m}^\sharp\,.\, \llparenthesis \mathbb{m}^\sharp = \bot^\sharp \ ?\ \emptyset \ :\ \{\mathbb{m} \in \mathbb{M} \mid \forall \mathbb{x} \in \mathbb{X}\,.\, \mathbb{m}(\mathbb{x}) \in \gamma^v(\mathbb{m}^\sharp(\mathbb{x}))\} \rrparenthesis$

Abstract partial order

$$\mathbb{m}_1^\sharp \subseteq^\sharp \mathbb{m}_2^\sharp \triangleq \forall x \in \mathbb{X} \,.\, \mathbb{m}_1^\sharp(x) \mathrel{\dot{\subseteq}_{\mathcal{V}}} \mathbb{m}_2^\sharp(x)$$

Abstract partial order

$$\mathbb{m}_1^\sharp \subseteq^\sharp \mathbb{m}_2^\sharp \triangleq \forall \mathrm{x} \in \mathbb{X} \,.\, \mathbb{m}_1^\sharp(\mathrm{x}) \mathrel{\dot{\subseteq}_\nu} \mathbb{m}_2^\sharp(\mathrm{x})$$

Abstract join and meet

$$\mathbb{m}_1^\sharp \cup^\sharp \mathbb{m}_2^\sharp \triangleq \lambda \mathrm{x} \,.\, \mathbb{m}_1^\sharp(\mathrm{x}) \cup^\nu \mathbb{m}_2^\sharp(\mathrm{x})$$

$$\mathbb{m}_1^\sharp \cap^\sharp \mathbb{m}_2^\sharp \triangleq \lambda \mathrm{x} \,.\, \mathbb{m}_1^\sharp(\mathrm{x}) \cap^\nu \mathbb{m}_2^\sharp(\mathrm{x})$$

Abstract partial order

$$\mathbb{m}_1^\sharp \subseteq^\sharp \mathbb{m}_2^\sharp \triangleq \forall \mathtt{x} \in \mathbb{X} \,.\, \mathbb{m}_1^\sharp(\mathtt{x}) \,\dot{\subseteq}_v\, \mathbb{m}_2^\sharp(\mathtt{x})$$

Abstract join and meet

$$\mathbb{m}_1^\sharp \cup^\sharp \mathbb{m}_2^\sharp \triangleq \lambda \mathtt{x} \,.\, \mathbb{m}_1^\sharp(\mathtt{x}) \cup^v \mathbb{m}_2^\sharp(\mathtt{x})$$
$$\mathbb{m}_1^\sharp \cap^\sharp \mathbb{m}_2^\sharp \triangleq \lambda \mathtt{x} \,.\, \mathbb{m}_1^\sharp(\mathtt{x}) \cap^v \mathbb{m}_2^\sharp(\mathtt{x})$$

Abstract transfer functions $\llparenthesis\, ^{\ell'} stmt \,\rrparenthesis^\sharp$ and $\mathbf{S}^\sharp_{stmt}$ on $\mathbb{M}^\sharp \to \mathbb{M}^\sharp$ derived from abstract arithmetic operations $-_v$, $+_v$, ...

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

Abstract assignments for relational domains

$$\llparenthesis\, {}^{\ell}\mathtt{x}=exp\, \rrparenthesis^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x}=exp}(\mathfrak{m}^{\sharp}) \triangleq \mathfrak{m}^{\sharp}[\mathtt{x} \hookleftarrow \textsc{AbsEval}[exp](\mathfrak{m}^{\sharp})]$$

Abstract assignments for relational domains

$$(\!|\ {}^{\ell}\mathtt{x}{=}exp\ |\!)^{\sharp}\mathbb{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x}=exp}(\mathbb{m}^{\sharp}) \triangleq \mathbb{m}^{\sharp}[\mathtt{x} \hookleftarrow \textsc{AbsEval}[exp](\mathbb{m}^{\sharp})]$$

Abstract evaluation for expressions $\textsc{AbsEval}[exp] : \mathbb{M}^{\sharp} \to \mathbb{M}^{\sharp}$

$$\begin{aligned}
\textsc{AbsEval}[exp]\bot^{\sharp} &\triangleq \bot^{v} \\
\textsc{AbsEval}[\mathtt{v}]\mathbb{m}^{\sharp} &\triangleq \alpha^{v}(\{v\}) \\
\textsc{AbsEval}[\mathtt{x}]\mathbb{m}^{\sharp} &\triangleq \mathbb{m}^{\sharp}(\mathtt{x}) \\
\textsc{AbsEval}[\text{-}\,exp]\mathbb{m}^{\sharp} &\triangleq \text{-}_{v}\,\textsc{AbsEval}[exp]\mathbb{m}^{\sharp} \\
\textsc{AbsEval}[exp_1 + exp_2]\mathbb{m}^{\sharp} &\triangleq \textsc{AbsEval}[exp_1]\mathbb{m}^{\sharp} +_{v} \textsc{AbsEval}[exp_2]\mathbb{m}^{\sharp} \\
&\ldots
\end{aligned}$$

Abstract assignments for relational domains

$$(\!|\,^{\ell}\texttt{x=}exp\,|\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\texttt{x=}exp}(\mathfrak{m}^{\sharp}) \triangleq \mathfrak{m}^{\sharp}[\texttt{x} \hookleftarrow \textsc{AbsEval}[exp](\mathfrak{m}^{\sharp})]$$

Abstract evaluation for expressions $\textsc{AbsEval}[exp] : \mathbb{M}^{\sharp} \to \mathbb{M}^{\sharp}$

$$\begin{aligned}
\textsc{AbsEval}[exp]\bot^{\sharp} &\triangleq \bot^{v} \\
\textsc{AbsEval}[\texttt{v}]\mathfrak{m}^{\sharp} &\triangleq \alpha^{v}(\{v\}) \\
\textsc{AbsEval}[\texttt{x}]\mathfrak{m}^{\sharp} &\triangleq \mathfrak{m}^{\sharp}(\texttt{x}) \\
\textsc{AbsEval}[\texttt{-}\,exp]\mathfrak{m}^{\sharp} &\triangleq \texttt{-}_{v}\,\textsc{AbsEval}[exp]\mathfrak{m}^{\sharp} \\
\textsc{AbsEval}[exp_1 + exp_2]\mathfrak{m}^{\sharp} &\triangleq \textsc{AbsEval}[exp_1]\mathfrak{m}^{\sharp} +_{v} \textsc{AbsEval}[exp_2]\mathfrak{m}^{\sharp} \\
&\ldots
\end{aligned}$$

Soundness: $\forall \mathfrak{m}^{\sharp} \in \mathbb{M}^{\sharp}\,.\,\{z \in \mathbb{Z} \mid z = \textsc{Eval}[exp](\mathfrak{m}) \wedge \mathfrak{m} \in \gamma^{v}_{nr}(\mathfrak{m}^{\sharp})\} \subseteq \gamma^{v}(\textsc{AbsEval}[exp](\mathfrak{m}^{\sharp}))$

Abstract transfer functions for conditions/boolean expressions $(\!|\,{}^{\ell'}bexp\,|\!)^\sharp$ and $\mathbf{S}^\sharp_{bexp}$ on $\mathbb{M}^\sharp \to \mathbb{M}^\sharp$

Abstract transfer functions for conditions/boolean expressions $(\!|\ {}^{\ell'}bexp\ |\!)^\sharp$ and $\mathbf{S}^\sharp_{bexp}$ on $\mathbb{M}^\sharp \to \mathbb{M}^\sharp$

- Abstract domain dependent (depend on $\mathbb{V}^\sharp$)

Abstract transfer functions for conditions/boolean expressions $(\!|\, {}^{\ell'} bexp \,|\!)^\sharp$ and $\mathbf{S}^\sharp_{bexp}$ on $\mathbb{M}^\sharp \to \mathbb{M}^\sharp$

- Abstract domain dependent (depend on $\mathbb{V}^\sharp$)
- The identity is a coarse but sound solution:

$$(\!|\, {}^{\ell'} bexp \,|\!)^\sharp \mathfrak{m}^\sharp \triangleq \mathbf{S}^\sharp_{bexp}(\mathfrak{m}^\sharp) \triangleq \mathfrak{m}^\sharp$$

Abstract transfer functions for conditions/boolean expressions $(\!|\, {}^{\ell'}bexp \,|\!)^\sharp$ and $\mathbf{S}^\sharp_{bexp}$ on $\mathbb{M}^\sharp \to \mathbb{M}^\sharp$

- Abstract domain dependent (depend on $\mathbb{V}^\sharp$)
- The identity is a coarse but sound solution:

$$(\!|\, {}^{\ell'}bexp \,|\!)^\sharp \mathfrak{m}^\sharp \triangleq \mathbf{S}^\sharp_{bexp}(\mathfrak{m}^\sharp) \triangleq \mathfrak{m}^\sharp$$

Iteration strategies and inductive cases are abstract domain independent

Abstract transfer functions for conditions/boolean expressions $\llparenthesis\,^{\ell'} bexp\,\rrparenthesis^\sharp$ and $\mathbf{S}^\sharp_{bexp}$ on $\mathbb{M}^\sharp \to \mathbb{M}^\sharp$

- Abstract domain dependent (depend on $\mathbb{V}^\sharp$)
- The identity is a coarse but sound solution:

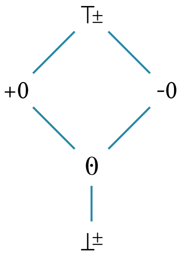
$$\llparenthesis\,^{\ell'} bexp\,\rrparenthesis^\sharp \mathfrak{m}^\sharp \triangleq \mathbf{S}^\sharp_{bexp}(\mathfrak{m}^\sharp) \triangleq \mathfrak{m}^\sharp$$

Iteration strategies and inductive cases are abstract domain independent

- $X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} \llparenthesis\,^{\ell'} stmt\,\rrparenthesis^\sharp X_{\ell'}$ such that $X_\ell \in \mathbb{M}^\sharp$
- $\mathbf{S}^\sharp_{\texttt{if}(bexp)\{stmt_1\}\texttt{else}\{stmt_2\}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{stmt_1} \circ \mathbf{S}^\sharp_{bexp}(\mathfrak{m}^\sharp) \cup^\sharp \mathbf{S}^\sharp_{stmt_2} \circ \mathbf{S}^\sharp_{\neg bexp}(\mathfrak{m}^\sharp)$

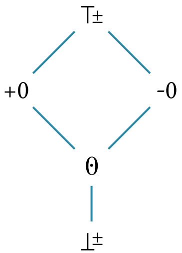
Non-relational Numerical Analyses

Complete lattice $\langle \mathbb{Z}^{\pm}, \subseteq^{\pm}, \cup^{\pm}, \cap^{\pm}, \bot^{\pm}, \top^{\pm} \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \xleftrightarrow[\alpha^{\pm}]{\gamma^{\pm}} \langle \mathbb{Z}^{\pm}, \subseteq^{\pm} \rangle$



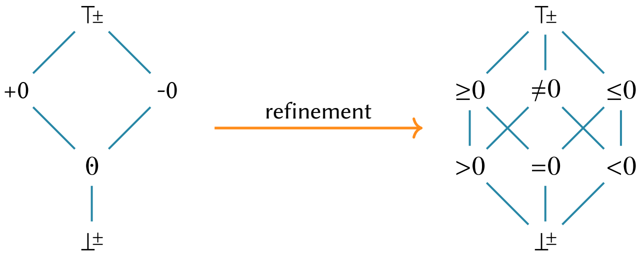
Sign abstract domain

Complete lattice $\langle \mathbb{Z}^{\pm}, \subseteq^{\pm}, \cup^{\pm}, \cap^{\pm}, \perp^{\pm}, \top^{\pm} \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha^{\pm}}{\overset{\gamma^{\pm}}{\leftrightarrows}} \langle \mathbb{Z}^{\pm}, \subseteq^{\pm} \rangle$



Abstraction $\alpha^{\pm}$ and concretization $\gamma^{\pm}$ defined as in the previous class

Complete lattice $\langle \mathbb{Z}^{\pm}, \subseteq^{\pm}, \cup^{\pm}, \cap^{\pm}, \bot^{\pm}, \top^{\pm} \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha^{\pm}}{\overset{\gamma^{\pm}}{\leftrightarrows}} \langle \mathbb{Z}^{\pm}, \subseteq^{\pm} \rangle$



Abstraction $\alpha^{\pm}$ and concretization $\gamma^{\pm}$ defined as in the previous class

Abstract arithmetic operations

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\pm(\{z\}) = \begin{cases} 0 & \text{if } z = 0 \\ \text{-}0 & \text{if } z < 0 \\ +0 & \text{if } z > 0 \end{cases}$$

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\pm(\{z\}) = \begin{cases} 0 & \text{if } z = 0 \\ \text{-}0 & \text{if } z < 0 \\ +0 & \text{if } z > 0 \end{cases}$$

$$\text{-}^\sharp\, \mathbb{v}^\pm \triangleq \alpha^\pm(\{\text{-}z \mid z \in \gamma^\pm(\mathbb{v}^\pm)\})$$

$$= \begin{cases} \bot^\pm & \text{if } \mathbb{v}^\pm = \bot^\pm \\ 0 & \text{if } \mathbb{v}^\pm = 0 \\ \text{-}0 & \text{if } \mathbb{v}^\pm = +0 \\ +0 & \text{if } \mathbb{v}^\pm = \text{-}0 \\ \top^\pm & \text{otherwise} \end{cases}$$

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\pm(\{z\}) = \begin{cases} 0 & \text{if } z = 0 \\ \text{-}0 & \text{if } z < 0 \\ +0 & \text{if } z > 0 \end{cases}$$

$$\text{-}^\sharp\, \mathbb{v}^\pm \triangleq \alpha^\pm(\{\text{-}z \mid z \in \gamma^\pm(\mathbb{v}^\pm)\}) = \begin{cases} \bot^\pm & \text{if } \mathbb{v}^\pm = \bot^\pm \\ 0 & \text{if } \mathbb{v}^\pm = 0 \\ \text{-}0 & \text{if } \mathbb{v}^\pm = +0 \\ +0 & \text{if } \mathbb{v}^\pm = \text{-}0 \\ \top^\pm & \text{otherwise} \end{cases}$$

$$\mathbb{v}_1^\pm +^\sharp \mathbb{v}_2^\pm \triangleq \alpha^\pm(\{z_1 + z_2 \mid z_1 \in \gamma^\pm(\mathbb{v}_1^\pm) \wedge z_2 \in \gamma^\pm(\mathbb{v}_2^\pm)\})$$

$$= \begin{cases} \bot^\pm & \text{if } \mathbb{v}_1^\pm = \bot^\pm \vee \mathbb{v}_2^\pm = \bot^\pm \\ 0 & \text{if } \mathbb{v}_1^\pm, \mathbb{v}_2^\pm \in \{0\} \\ \text{-}0 & \text{if } \mathbb{v}_1^\pm, \mathbb{v}_2^\pm \in \{0, \text{-}0\} \\ +0 & \text{if } \mathbb{v}_1^\pm, \mathbb{v}_2^\pm \in \{0, +0\} \\ \top^\pm & \text{otherwise} \end{cases}$$

Abstract conditions/boolean expressions for $\mathbb{Z}^{\pm}$

Abstract conditions/boolean expressions for $\mathbb{Z}^{\pm}$

$$(\!|\, {}^{\ell}\texttt{x <= 0} \,|\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\texttt{x<=0}}(\mathfrak{m}^{\sharp}) \triangleq \begin{cases} \mathfrak{m}^{\sharp}[\mathtt{x} \hookleftarrow 0] & \text{if } \mathfrak{m}^{\sharp}(\mathtt{x}) \in \{0, +0\} \\ \mathfrak{m}^{\sharp}[\mathtt{x} \hookleftarrow -0] & \text{if } \mathfrak{m}^{\sharp}(\mathtt{x}) \in \{-0, \top^{\pm}\} \\ \bot^{\pm} & \text{otherwise} \end{cases}$$

Abstract conditions/boolean expressions for $\mathbb{Z}^{\pm}$

$$
(\!|\, {}^{\ell}\texttt{x <= 0} \,|\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\texttt{x<=0}}(\mathfrak{m}^{\sharp}) \triangleq \begin{cases} \mathfrak{m}^{\sharp}[\texttt{x} \leftarrow \mathbb{0}] & \text{if } \mathfrak{m}^{\sharp}(\texttt{x}) \in \{\mathbb{0}, +0\} \\ \mathfrak{m}^{\sharp}[\texttt{x} \leftarrow -0] & \text{if } \mathfrak{m}^{\sharp}(\texttt{x}) \in \{-0, \top\pm\} \\ \bot^{\pm} & \text{otherwise} \end{cases}
$$

$$
(\!|\, {}^{\ell}\texttt{x <= } z \,|\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\texttt{x<=}z}(\mathfrak{m}^{\sharp}) \triangleq \begin{cases} (\!|\, {}^{\ell}\texttt{x <= 0} \,|\!)^{\sharp}\mathfrak{m}^{\sharp} = \mathbf{S}^{\sharp}_{\texttt{x<=0}}(\mathfrak{m}^{\sharp}) & \text{if } z \leq 0 \\ \mathfrak{m}^{\sharp} & \text{otherwise} \end{cases}
$$

Abstract conditions/boolean expressions for $\mathbb{Z}^{\pm}$

$$(\!|\, {}^{\ell}\mathtt{x\ <=\ 0}\, |\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x<=0}}(\mathfrak{m}^{\sharp}) \triangleq \begin{cases} \mathfrak{m}^{\sharp}[\mathtt{x} \hookleftarrow 0] & \text{if } \mathfrak{m}^{\sharp}(\mathtt{x}) \in \{0, +0\} \\ \mathfrak{m}^{\sharp}[\mathtt{x} \hookleftarrow -0] & \text{if } \mathfrak{m}^{\sharp}(\mathtt{x}) \in \{-0, \top\pm\} \\ \bot^{\pm} & \text{otherwise} \end{cases}$$

$$(\!|\, {}^{\ell}\mathtt{x\ <=\ } z\, |\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x<=}z}(\mathfrak{m}^{\sharp}) \triangleq \begin{cases} (\!|\, {}^{\ell}\mathtt{x\ <=\ 0}\, |\!)^{\sharp}\mathfrak{m}^{\sharp} = \mathbf{S}^{\sharp}_{\mathtt{x<=0}}(\mathfrak{m}^{\sharp}) & \text{if } z \leq 0 \\ \mathfrak{m}^{\sharp} & \text{otherwise} \end{cases}$$

$$\begin{aligned} (\!|\, {}^{\ell}\mathtt{x\ <=\ y}\, |\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x<=y}}(\mathfrak{m}^{\sharp}) \triangleq\ & (\!|\, \mathfrak{m}^{\sharp}(\mathtt{y}) \in \{0, -0\} \mathrel{?} (\!|\, {}^{\ell}\mathtt{x\ <=\ 0}\, |\!)^{\sharp}\mathfrak{m}^{\sharp} = \mathbf{S}^{\sharp}_{\mathtt{x<=0}}(\mathfrak{m}^{\sharp}) \mathbin{\text{⁝}} \mathfrak{m}^{\sharp} \,|\!) \\ & \cap^{\sharp} \\ & (\!|\, \mathfrak{m}^{\sharp}(\mathtt{x}) \in \{0, +0\} \mathrel{?} (\!|\, {}^{\ell}\mathtt{y\ >=\ 0}\, |\!)^{\sharp}\mathfrak{m}^{\sharp} = \mathbf{S}^{\sharp}_{\mathtt{y>=0}}(\mathfrak{m}^{\sharp}) \mathbin{\text{⁝}} \mathfrak{m}^{\sharp} \,|\!) \end{aligned}$$

Abstract conditions/boolean expressions for $\mathbb{Z}^{\pm}$ identity for the other cases

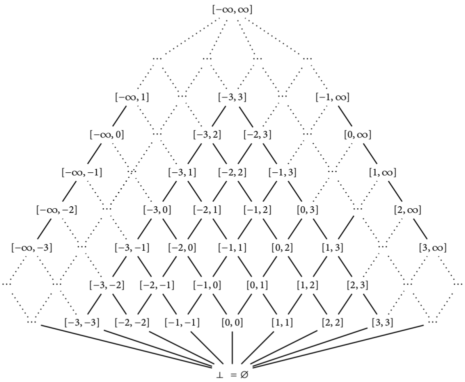
$$(\!|\, {}^{\ell}\mathtt{x} \;\mathtt{<=}\; \mathtt{0} \,|\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x<=0}}(\mathfrak{m}^{\sharp}) \triangleq \begin{cases} \mathfrak{m}^{\sharp}[\mathtt{x} \leftarrow 0] & \text{if } \mathfrak{m}^{\sharp}(\mathtt{x}) \in \{0, +0\} \\ \mathfrak{m}^{\sharp}[\mathtt{x} \leftarrow \text{-}0] & \text{if } \mathfrak{m}^{\sharp}(\mathtt{x}) \in \{\text{-}0, \top^{\pm}\} \\ \bot^{\pm} & \text{otherwise} \end{cases}$$

$$(\!|\, {}^{\ell}\mathtt{x} \;\mathtt{<=}\; z \,|\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x<=}z}(\mathfrak{m}^{\sharp}) \triangleq \begin{cases} (\!|\, {}^{\ell}\mathtt{x} \;\mathtt{<=}\; \mathtt{0} \,|\!)^{\sharp}\mathfrak{m}^{\sharp} = \mathbf{S}^{\sharp}_{\mathtt{x<=0}}(\mathfrak{m}^{\sharp}) & \text{if } z \leq 0 \\ \mathfrak{m}^{\sharp} & \text{otherwise} \end{cases}$$

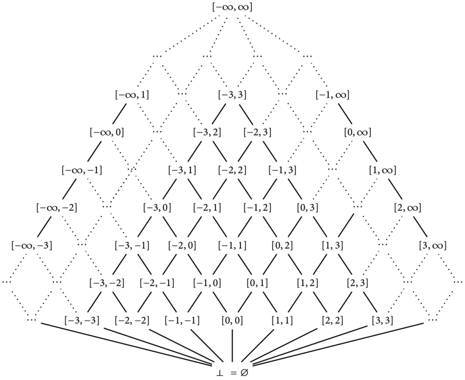
$$\begin{aligned} (\!|\, {}^{\ell}\mathtt{x} \;\mathtt{<=}\; \mathtt{y} \,|\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x<=y}}(\mathfrak{m}^{\sharp}) \triangleq\; & [\![\, \mathfrak{m}^{\sharp}(\mathtt{y}) \in \{0, \text{-}0\} \;?\; (\!|\, {}^{\ell}\mathtt{x} \;\mathtt{<=}\; \mathtt{0} \,|\!)^{\sharp}\mathfrak{m}^{\sharp} = \mathbf{S}^{\sharp}_{\mathtt{x<=0}}(\mathfrak{m}^{\sharp}) \;\text{:}\; \mathfrak{m}^{\sharp} \,]\!] \\ & \cap^{\sharp} \\ & [\![\, \mathfrak{m}^{\sharp}(\mathtt{x}) \in \{0, +0\} \;?\; (\!|\, {}^{\ell}\mathtt{y} \;\mathtt{>=}\; \mathtt{0} \,|\!)^{\sharp}\mathfrak{m}^{\sharp} = \mathbf{S}^{\sharp}_{\mathtt{y>=0}}(\mathfrak{m}^{\sharp}) \;\text{:}\; \mathfrak{m}^{\sharp} \,]\!] \end{aligned}$$

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

Complete lattice $\langle \mathbb{Z}^\imath, \subseteq^\imath, \cup^\imath, \cap^\imath, \perp^\imath, \top^\imath \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha^\imath}{\overset{\gamma^\imath}{\leftrightarrows}} \langle \mathbb{Z}^\imath, \subseteq^\imath \rangle$



Complete lattice $\langle \mathbb{Z}^\imath, \subseteq^\imath, \cup^\imath, \cap^\imath, \perp^\imath, \top^\imath \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha^\imath}{\overset{\gamma^\imath}{\leftrightarrows}} \langle \mathbb{Z}^\imath, \subseteq^\imath \rangle$



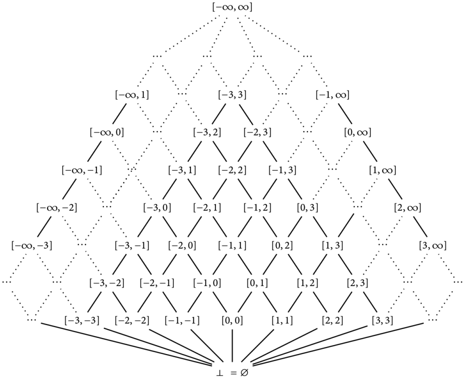
$$[a,b] \subseteq^\imath [a',b'] \triangleq a \geq a' \wedge b \leq b'$$

$$[a,b] \cup^\imath [a',b'] \triangleq [a \curlyvee a', b \curlywedge b']$$

$$[a,b] \cap^\imath [a',b'] \triangleq \begin{cases} [a \curlywedge a', b \curlyvee b'] & \text{if } a \curlywedge a' \leq b \curlyvee b' \\ \perp^\imath & \text{otherwise} \end{cases}$$

$\perp^\imath \triangleq [\infty, -\infty]$ and $\top^\imath \triangleq [-\infty, \infty]$

Complete lattice $\langle \mathbb{Z}^\imath, \subseteq^\imath, \cup^\imath, \cap^\imath, \bot^\imath, \top^\imath \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha^\imath}{\overset{\gamma^\imath}{\leftrightarrows}} \langle \mathbb{Z}^\imath, \subseteq^\imath \rangle$



$$[a,b] \subseteq^\imath [a',b'] \triangleq a \geq a' \wedge b \leq b'$$

$$[a,b] \cup^\imath [a',b'] \triangleq [a \vee a', b \wedge b']$$

$$[a,b] \cap^\imath [a',b'] \triangleq \begin{cases} [a \wedge a', b \vee b'] & \text{if } a \wedge a' \leq b \vee b' \\ \bot^\imath & \text{otherwise} \end{cases}$$

$\bot^\imath \triangleq [\infty, -\infty]$ and $\top^\imath \triangleq [-\infty, \infty]$

Abstraction $\alpha^\imath$ and concretization $\gamma^\imath$ defined as in the previous class

Abstract arithmetic operations

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\imath(\{z\}) = [z, z]$$

UNIVERSITÀ di VERONA Dipartimento di INFORMATICA

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\imath(\{z\}) = [z, z]$$

$$-^\sharp [a, a'] \triangleq \alpha^\imath(\{-z \mid z \in \gamma^\imath([a, a'])\}) = [-a', -a]$$

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\imath(\{z\}) = [z, z]$$

$$-^\sharp [a, a'] \triangleq \alpha^\imath(\{-z \mid z \in \gamma^\imath([a, a'])\}) = [-a', -a]$$

$$\begin{aligned}
[a, a'] +^\sharp [b, b'] &\triangleq \alpha^\imath(\{z_1 + z_2 \mid z_1 \in \gamma^\imath([a, a']) \wedge z_2 \in \gamma^\imath([b, b'])\}) = [a + b, a' + b'] \\
[a, a'] -^\sharp [b, b'] &\triangleq \alpha^\imath(\{z_1 - z_2 \mid z_1 \in \gamma^\imath([a, a']) \wedge z_2 \in \gamma^\imath([b, b'])\}) = [a - b', a' - b] \\
[a, a'] \times^\sharp [b, b'] &\triangleq \alpha^\imath(\{z_1 \cdot z_2 \mid z_1 \in \gamma^\imath([a, a']) \wedge z_2 \in \gamma^\imath([b, b'])\}) \\
&= [\bigvee\{a \cdot b,\ a \cdot b',\ a' \cdot b,\ a' \cdot b'\}, \bigwedge\{a \cdot b,\ a \cdot b',\ a' \cdot b,\ a' \cdot b'\}]
\end{aligned}$$

Abstract arithmetic operations

$z^\sharp \triangleq \alpha^\imath(\{z\}) = [z, z]$

Operations are strict: $\text{-}^\sharp \bot^\imath \triangleq \bot^\imath$, $[a, a'] +^\sharp \bot^\imath \triangleq \bot^\imath$, ...

$\text{-}^\sharp\, [a, a'] \triangleq \alpha^\imath(\{\text{-}\,z \mid z \in \gamma^\imath([a, a'])\}) = [\text{-}a', \text{-}a]$

$$
\begin{aligned}
[a, a'] +^\sharp [b, b'] &\triangleq \alpha^\imath(\{z_1 + z_2 \mid z_1 \in \gamma^\imath([a, a']) \wedge z_2 \in \gamma^\imath([b, b'])\}) = [a + b, a' + b'] \\
[a, a'] -^\sharp [b, b'] &\triangleq \alpha^\imath(\{z_1 - z_2 \mid z_1 \in \gamma^\imath([a, a']) \wedge z_2 \in \gamma^\imath([b, b'])\}) = [a - b', a' - b] \\
[a, a'] \times^\sharp [b, b'] &\triangleq \alpha^\imath(\{z_1 \cdot z_2 \mid z_1 \in \gamma^\imath([a, a']) \wedge z_2 \in \gamma^\imath([b, b'])\}) \\
&= [\textstyle\bigvee\{a \cdot b,\ a \cdot b',\ a' \cdot b,\ a' \cdot b'\}, \bigwedge\{a \cdot b,\ a \cdot b',\ a' \cdot b,\ a' \cdot b'\}]
\end{aligned}
$$

Abstract conditions/boolean expressions for $\mathbb{Z}^\imath$

Abstract conditions/boolean expressions for $\mathbb{Z}^\imath$

Assuming $\mathfrak{m}^\sharp(\mathtt{x}) = [a, a']$

$$(\!|\, {}^{\ell}\mathtt{x} \;\; \texttt{<=}\; z \,|\!)^\sharp \mathfrak{m}^\sharp \triangleq \mathbf{S}^\sharp_{\mathtt{x} \texttt{<=} z}(\mathfrak{m}^\sharp) \triangleq \begin{cases} \mathfrak{m}^\sharp_\perp & \text{if } a > z \\ \mathfrak{m}^\sharp[\mathtt{x} \hookleftarrow [a, a' \curlywedge z]] & \text{otherwise} \end{cases}$$

Abstract conditions/boolean expressions for $\mathbb{Z}^\imath$

Assuming $\mathfrak{m}^\sharp(\mathtt{x}) = [a, a']$

$$(\!|\, {}^{\ell}\mathtt{x} \;\texttt{<=}\; z \,|\!)^\sharp \mathfrak{m}^\sharp \triangleq \mathbf{S}^\sharp_{\mathtt{x<=}z}(\mathfrak{m}^\sharp) \triangleq \begin{cases} \mathfrak{m}^\sharp_\perp & \text{if } a > z \\ \mathfrak{m}^\sharp[\mathtt{x} \leftarrow [a, a' \curlywedge z]] & \text{otherwise} \end{cases}$$

Assuming $\mathfrak{m}^\sharp(\mathtt{x}) = [a, a']$ and $\mathfrak{m}^\sharp(\mathtt{y}) = [b, b']$

$$(\!|\, {}^{\ell}\mathtt{x} \;\texttt{<=}\; \mathtt{y} \,|\!)^\sharp \mathfrak{m}^\sharp \triangleq \mathbf{S}^\sharp_{\mathtt{x<=y}}(\mathfrak{m}^\sharp) \triangleq \begin{cases} \mathfrak{m}^\sharp_\perp & \text{if } a > b' \\ \mathfrak{m}^\sharp[\mathtt{x} \leftarrow [a, a' \curlywedge b']\ \mathtt{y} \leftarrow [a \curlyvee b, b']] & \text{otherwise} \end{cases}$$

Abstract conditions/boolean expressions for $\mathbb{Z}^\imath$ identity for the other cases

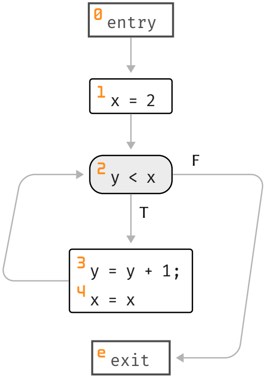
Assuming $\mathbb{m}^\sharp(\mathtt{x}) = [a, a']$

$$(\!|\, {}^{\ell}\mathtt{x}\ \texttt{<=}\ z \,|\!)^\sharp \mathbb{m}^\sharp \triangleq \mathbf{S}^\sharp_{\mathtt{x}\texttt{<=}z}(\mathbb{m}^\sharp) \triangleq \begin{cases} \mathbb{m}^\sharp_\bot & \text{if } a > z \\ \mathbb{m}^\sharp[\mathtt{x} \hookleftarrow [a, a' \curlyvee z]] & \text{otherwise} \end{cases}$$

Assuming $\mathbb{m}^\sharp(\mathtt{x}) = [a, a']$ and $\mathbb{m}^\sharp(\mathtt{y}) = [b, b']$

$$(\!|\, {}^{\ell}\mathtt{x}\ \texttt{<=}\ \mathtt{y} \,|\!)^\sharp \mathbb{m}^\sharp \triangleq \mathbf{S}^\sharp_{\mathtt{x}\texttt{<=}\mathtt{y}}(\mathbb{m}^\sharp) \triangleq \begin{cases} \mathbb{m}^\sharp_\bot & \text{if } a > b' \\ \mathbb{m}^\sharp[\mathtt{x} \hookleftarrow [a, a' \curlyvee b']\ \mathtt{y} \hookleftarrow [a \curlywedge b, b']] & \text{otherwise} \end{cases}$$

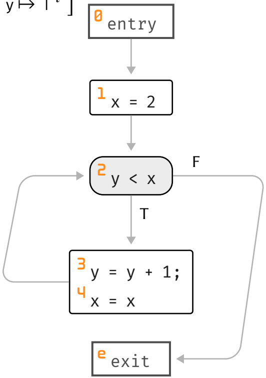
Intervals analysis: example



Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\imath$

Intervals analysis: example

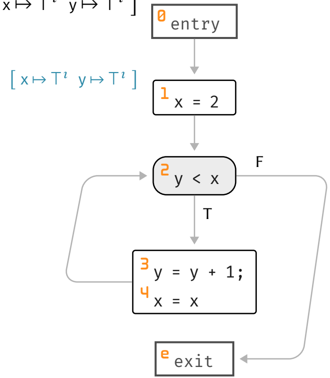
Input: $[\,x \mapsto \top^{\imath} \quad y \mapsto \top^{\imath}\,]$



Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^{\imath}$

Intervals analysis: example

Input: $[\, x \mapsto \top^{\imath} \;\; y \mapsto \top^{\imath} \,]$

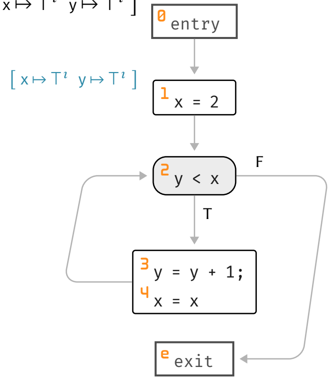


Abstract memories: $\mathbb{M}^{\sharp} = \mathbb{X} \to \mathbb{Z}^{\imath}$

Intervals analysis: example

$\mathbf{S}^\sharp_{\texttt{x=2}}(\mathfrak{m}^\sharp) \triangleq \mathfrak{m}^\sharp[\texttt{x} \hookleftarrow \alpha^\imath(2)]$

Input: $[\,\texttt{x} \mapsto \top^\imath \;\; \texttt{y} \mapsto \top^\imath\,]$

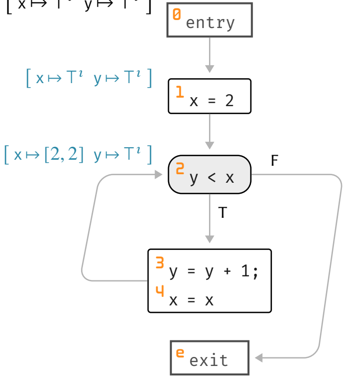


Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\imath$

Intervals analysis: example

$$\mathbf{S}^{\sharp}_{\mathtt{x=2}}(\mathfrak{m}^{\sharp}) \triangleq \mathfrak{m}^{\sharp}[\mathtt{x} \leftarrow \alpha^{\imath}(2)] = [\,\mathtt{x} \mapsto [2,2] \;\; \mathtt{y} \mapsto \top^{\imath}\,]$$

Input: $[\,\mathtt{x} \mapsto \top^{\imath} \;\; \mathtt{y} \mapsto \top^{\imath}\,]$

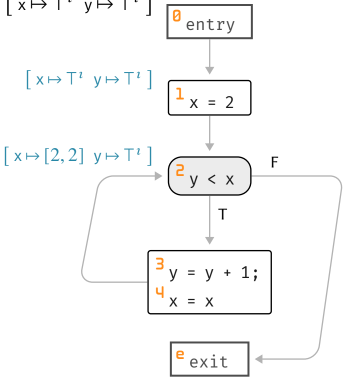


Abstract memories: $\mathbb{M}^{\sharp} = \mathbb{X} \to \mathbb{Z}^{\imath}$

Intervals analysis: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}\left(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda \overline{\mathfrak{m}}^\sharp .\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathfrak{m}}^\sharp)\right)$$

Input: $[\, x \mapsto \top^\imath \quad y \mapsto \top^\imath \,]$

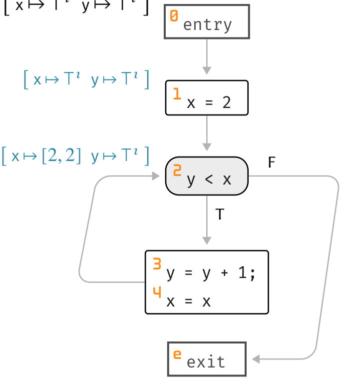


Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\imath$

Intervals analysis: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathfrak{m}}^\sharp.\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathfrak{m}}^\sharp))$$

Input: $[\,\texttt{x} \mapsto \top^\imath\ \ \texttt{y} \mapsto \top^\imath\,]$



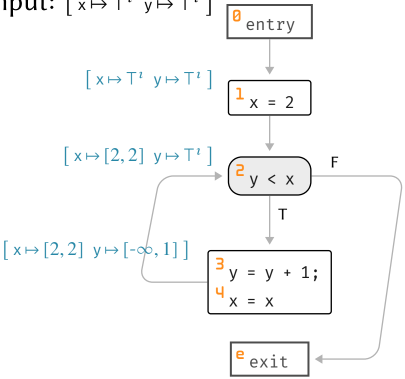
$X^0 = [\,\texttt{x} \mapsto [2,2]\ \ \texttt{y} \mapsto \top^\imath\,]$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\imath$

Intervals analysis: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(m^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda \overline{m}^\sharp .\ m^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{m}^\sharp))$$

Input: $[\,x \mapsto \top^\imath \;\; y \mapsto \top^\imath\,]$

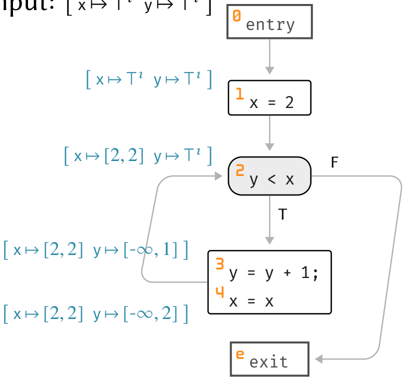


$$X^0 = [\,x \mapsto [2,2] \;\; y \mapsto \top^\imath\,]$$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\imath$

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}\big(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathfrak{m}}^\sharp\ .\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathfrak{m}}^\sharp)\big)$$

Input: $[\,\mathtt{x} \mapsto \top^\imath\ \ \mathtt{y} \mapsto \top^\imath\,]$



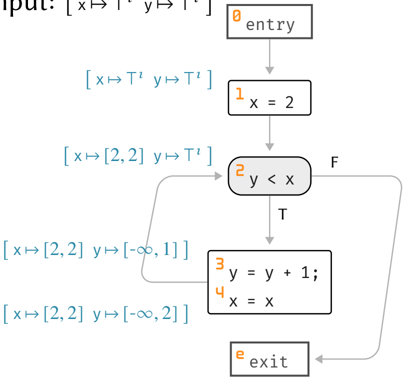
$$X^0 = [\,\mathtt{x} \mapsto [2,2]\ \ \mathtt{y} \mapsto \top^\imath\,]$$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\imath$

Intervals analysis: example

$$\mathbf{S}^\sharp_{\mathtt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\mathtt{x>=y}}\left(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathfrak{m}}^\sharp .\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\mathtt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\mathtt{x<y}}(\overline{\mathfrak{m}}^\sharp)\right)$$

Input: $[\,\mathtt{x} \mapsto \top^\imath\ \ \mathtt{y} \mapsto \top^\imath\,]$



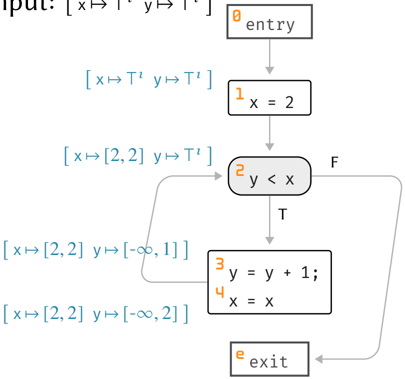
$$X^0 = [\,\mathtt{x} \mapsto [2,2]\ \ \mathtt{y} \mapsto \top^\imath\,]$$
$$X^1 = [\,\mathtt{x} \mapsto [2,2]\ \ \mathtt{y} \mapsto \top^\imath\,]$$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\imath$

Intervals analysis: example

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}\left(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda \overline{\mathfrak{m}}^\sharp .\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathfrak{m}}^\sharp)\right)$$

Input: $[\,\texttt{x} \mapsto \top^\imath\ \ \texttt{y} \mapsto \top^\imath\,]$



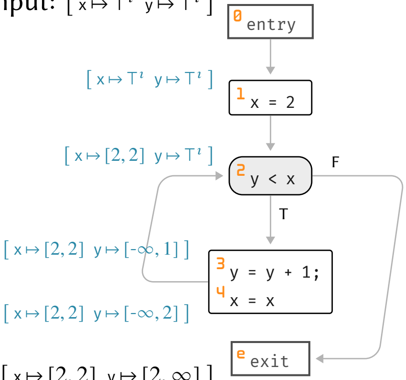
$$X^0 = [\,\texttt{x} \mapsto [2,2]\ \ \texttt{y} \mapsto \top^\imath\,]$$

$$\mathrm{lfp} = [\,\texttt{x} \mapsto [2,2]\ \ \texttt{y} \mapsto \top^\imath\,]$$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\imath$

$$\mathbf{S}^\sharp_{\texttt{while(x<y)\{y=y+1;x=x\}}}(\mathfrak{m}^\sharp) \triangleq \mathbf{S}^\sharp_{\texttt{x>=y}}\left(\mathrm{lfp}^{\subseteq^\sharp}\ \lambda\overline{\mathfrak{m}}^\sharp\,.\ \mathfrak{m}^\sharp \cup^\sharp \mathbf{S}^\sharp_{\texttt{y=y+1;x=x}} \circ \mathbf{S}^\sharp_{\texttt{x<y}}(\overline{\mathfrak{m}}^\sharp)\right)$$

Input: $[\, \texttt{x} \mapsto \top^\imath \ \ \texttt{y} \mapsto \top^\imath \,]$



Output: $[\, \texttt{x} \mapsto [2,2] \ \ \texttt{y} \mapsto [2,\infty] \,]$

$X^0 = [\, \texttt{x} \mapsto [2,2] \ \ \texttt{y} \mapsto \top^\imath \,]$

$\mathrm{lfp} = [\, \texttt{x} \mapsto [2,2] \ \ \texttt{y} \mapsto \top^\imath \,]$

Output: $[\, \texttt{x} \mapsto [2,2] \ \ \texttt{y} \mapsto [2,\infty] \,]$

Abstract memories: $\mathbb{M}^\sharp = \mathbb{X} \rightarrow \mathbb{Z}^\imath$

The abstract Post-conditions semantics for intervals is sound:

$$[\![P]\!]^{\mathsf{c}} \circ \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp}) \subseteq \gamma^{\imath}_{nr} \circ [\![P]\!]^{\sharp}\mathbb{m}^{\sharp} \qquad \mathbb{m}^{\sharp} \triangleq \alpha^{\imath}_{nr}(\mathfrak{I} \subseteq \mathbb{M}) \qquad \alpha^{\imath}_{nr} \circ [\![P]\!]^{\mathsf{c}}\mathfrak{I} \subseteq^{\sharp} [\![P]\!]^{\sharp} \circ \alpha^{\imath}_{nr}(\mathfrak{I})$$

The abstract Post-conditions semantics for intervals is sound:

$$\llbracket P \rrbracket^{c} \circ \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp}) \subseteq \gamma^{\imath}_{nr} \circ \llbracket P \rrbracket^{\sharp}\mathbb{m}^{\sharp} \qquad \mathbb{m}^{\sharp} \triangleq \alpha^{\imath}_{nr}(\mathfrak{I} \subseteq \mathbb{M}) \qquad \alpha^{\imath}_{nr} \circ \llbracket P \rrbracket^{c}\mathfrak{I} \subseteq^{\sharp} \llbracket P \rrbracket^{\sharp} \circ \alpha^{\imath}_{nr}(\mathfrak{I})$$

(proof)

Prove $\alpha^{\imath}_{nr} \circ \llbracket P \rrbracket^{c} \circ \gamma^{\imath}_{nr} \subseteq^{\sharp} \llbracket P \rrbracket^{\sharp}$ by structural induction on $P = stmt_1 \,;\, \ldots \,;\, stmt_n$

The abstract Post-conditions semantics for intervals is sound:

$$[\![P]\!]^{c} \circ \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp}) \subseteq \gamma^{\imath}_{nr} \circ [\![P]\!]^{\sharp}\mathbb{m}^{\sharp} \qquad \mathbb{m}^{\sharp} \triangleq \alpha^{\imath}_{nr}(\mathfrak{I} \subseteq \mathbb{M}) \qquad \alpha^{\imath}_{nr} \circ [\![P]\!]^{c}\mathfrak{I} \subseteq^{\sharp} [\![P]\!]^{\sharp} \circ \alpha^{\imath}_{nr}(\mathfrak{I})$$

(proof)

Prove $\alpha^{\imath}_{nr} \circ [\![P]\!]^{c} \circ \gamma^{\imath}_{nr} \subseteq^{\sharp} [\![P]\!]^{\sharp}$ by structural induction on $P = stmt_1 \,;\, \dots \,;\, stmt_n$

Recall that

- $[\![P]\!]^{c}\mathfrak{I} = \mathbf{S}^{c}_{stmt_n} \circ \dots \circ \mathbf{S}^{c}_{stmt_1}(\mathfrak{I})$ and $[\![P]\!]^{\sharp}\mathbb{m}^{\sharp} = \mathbf{S}^{\sharp}_{stmt_n} \circ \dots \circ \mathbf{S}^{\sharp}_{stmt_1}(\mathbb{m}^{\sharp})$

The abstract Post-conditions semantics for intervals is sound:

$$[\![P]\!]^{c} \circ \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp}) \subseteq \gamma^{\imath}_{nr} \circ [\![P]\!]^{\sharp}\mathbb{m}^{\sharp} \qquad \mathbb{m}^{\sharp} \triangleq \alpha^{\imath}_{nr}(\mathfrak{I} \subseteq \mathbb{M}) \qquad \alpha^{\imath}_{nr} \circ [\![P]\!]^{c}\mathfrak{I} \subseteq^{\sharp} [\![P]\!]^{\sharp} \circ \alpha^{\imath}_{nr}(\mathfrak{I})$$

(proof)

Prove $\alpha^{\imath}_{nr} \circ [\![P]\!]^{c} \circ \gamma^{\imath}_{nr} \subseteq^{\sharp} [\![P]\!]^{\sharp}$ by structural induction on $P = stmt_1 ; \ldots ; stmt_n$

Recall that

- $[\![P]\!]^{c}\mathfrak{I} = \mathbf{S}^{c}_{stmt_n} \circ \ldots \circ \mathbf{S}^{c}_{stmt_1}(\mathfrak{I})$ and $[\![P]\!]^{\sharp}\mathbb{m}^{\sharp} = \mathbf{S}^{\sharp}_{stmt_n} \circ \ldots \circ \mathbf{S}^{\sharp}_{stmt_1}(\mathbb{m}^{\sharp})$
- We assume soundness of arithmetic operations, and hence of AbsEval$[exp]$

$$\forall \mathbb{m}^{\sharp} \in \mathbb{M}^{\sharp} \,.\, \{z \in \mathbb{Z} \mid z = \textsc{Eval}[exp](\mathbb{m}) \wedge \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp})\} \subseteq \gamma^{\imath}(\textsc{AbsEval}[exp](\mathbb{m}^{\sharp}))$$

(proof) Base case of assignments

$\alpha^{\imath}_{nr} \circ \mathbf{S}^{c}_{\mathtt{x}=exp} \circ \gamma^{\imath}_{nr}(\mathfrak{m}^{\sharp})$

(proof) Base case of assignments

$$\alpha^{\imath}_{nr} \circ \mathbf{S}^{c}_{\mathtt{x}=exp} \circ \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp})$$

$= $ by definition of $\mathbf{S}^{c}_{\cdot}$

$$\alpha^{\imath}_{nr}(\{\mathbb{m}[\mathtt{x} \hookleftarrow z] \mid \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp}) \wedge z = \textsc{Eval}[exp](\mathbb{m})\})$$

(proof) Base case of assignments

$\alpha^{\imath}_{nr} \circ \mathbf{S}^{c}_{\mathtt{x}=exp} \circ \gamma^{\imath}_{nr}(\mathfrak{m}^{\sharp})$

$=$ by definition of $\mathbf{S}^{c}_{\cdot}$

$\alpha^{\imath}_{nr}(\{\mathfrak{m}[\mathtt{x} \hookleftarrow z] \mid \mathfrak{m} \in \gamma^{\imath}_{nr}(\mathfrak{m}^{\sharp}) \wedge z = \textsc{Eval}[exp](\mathfrak{m})\})$

$=$ by $\alpha^{\imath}_{nr} = \dot{\alpha}^{\imath} \circ \alpha_{nr}$ and definition of α_{nr}

$\dot{\alpha}^{\imath} \circ (\lambda \mathtt{y}\,.\, \{\mathfrak{m}(\mathtt{y}) \mid \mathfrak{m} \in \{\mathfrak{m}[\mathtt{x} \hookleftarrow z] \mid \mathfrak{m} \in \gamma^{\imath}_{nr}(\mathfrak{m}^{\sharp}) \wedge z = \textsc{Eval}[exp](\mathfrak{m})\}\})$

(proof) Base case of assignments

$\alpha^{\imath}_{nr} \circ \mathbf{S}^{c}_{\mathtt{x}=exp} \circ \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp})$

$=$ by definition of $\mathbf{S}^{c}_{.}$

$\alpha^{\imath}_{nr}(\{\mathbb{m}[\mathtt{x} \hookleftarrow z] \mid \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp}) \wedge z = \text{EVAL}[exp](\mathbb{m})\})$

$=$ by $\alpha^{\imath}_{nr} = \dot{\alpha}^{\imath} \circ \alpha_{nr}$ and definition of α_{nr}

$\dot{\alpha}^{\imath} \circ (\lambda \mathtt{y} .\ \{\mathbb{m}(\mathtt{y}) \mid \mathbb{m} \in \{\mathbb{m}[\mathtt{x} \hookleftarrow z] \mid \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp}) \wedge z = \text{EVAL}[exp](\mathbb{m})\}\})$

$=$

$\dot{\alpha}^{\imath} \circ (\lambda \mathtt{y} .\ \llbracket \mathtt{y} = \mathtt{x}\ ?\ \{z \mid \exists \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp}) .\, z = \text{EVAL}[exp](\mathbb{m})\} : \{\mathbb{m}(\mathtt{y}) \mid \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp})\} \rrbracket)$

(proof) Base case of assignments

$\alpha^{\imath}_{nr} \circ \mathbf{S}^{c}_{\mathtt{x}=exp} \circ \gamma^{\imath}_{nr}(\mathtt{m}^{\sharp})$

$=$ by definition of $\mathbf{S}^{c}_{\cdot}$

$\alpha^{\imath}_{nr}(\{\mathtt{m}[\mathtt{x} \hookleftarrow z] \mid \mathtt{m} \in \gamma^{\imath}_{nr}(\mathtt{m}^{\sharp}) \wedge z = \textsc{Eval}[exp](\mathtt{m})\})$

$=$ by $\alpha^{\imath}_{nr} = \dot{\alpha}^{\imath} \circ \alpha_{nr}$ and definition of α_{nr}

$\dot{\alpha}^{\imath} \circ (\lambda \mathtt{y}\,.\, \{\mathtt{m}(\mathtt{y}) \mid \mathtt{m} \in \{\mathtt{m}[\mathtt{x} \hookleftarrow z] \mid \mathtt{m} \in \gamma^{\imath}_{nr}(\mathtt{m}^{\sharp}) \wedge z = \textsc{Eval}[exp](\mathtt{m})\}\})$

$=$

$\dot{\alpha}^{\imath} \circ (\lambda \mathtt{y}\,.\, (\!\vert\, \mathtt{y} = \mathtt{x} \,?\, \{z \mid \exists \mathtt{m} \in \gamma^{\imath}_{nr}(\mathtt{m}^{\sharp})\,.\, z = \textsc{Eval}[exp](\mathtt{m})\} \,{\circ\atop\circ}\, \{\mathtt{m}(\mathtt{y}) \mid \mathtt{m} \in \gamma^{\imath}_{nr}(\mathtt{m}^{\sharp})\} \,\vert\!))$

$\subseteq^{\sharp}$ by soundness of $\textsc{AbsEval}[exp]$

$\dot{\alpha}^{\imath} \circ (\lambda \mathtt{y}\,.\, (\!\vert\, \mathtt{y} = \mathtt{x} \,?\, \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathtt{m}^{\sharp}) \,{\circ\atop\circ}\, \{\mathtt{m}(\mathtt{y}) \mid \mathtt{m} \in \gamma^{\imath}_{nr}(\mathtt{m}^{\sharp})\} \,\vert\!))$

(proof) Cont'd

$$\dot{\alpha}^{\imath} \circ (\lambda y\,.\, (\!| y = x \mathrel{?} \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathfrak{m}^{\sharp}) \mathrel{\mathring{\circ}} \{\mathfrak{m}(y) \mid \mathfrak{m} \in \gamma^{\imath}_{nr}(\mathfrak{m}^{\sharp})\} |\!))$$

(proof) Cont'd

$$\dot{\alpha}^{\imath} \circ (\lambda \mathsf{y}\,.\, [\![\, \mathsf{y} = \mathsf{x}\ ?\ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathbb{m}^{\sharp}) : \{\mathbb{m}(\mathsf{y}) \mid \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp})\}\,]\!])$$

$=$ by definition of $\dot{\alpha}^{\imath}$

$$\lambda \mathsf{y}\,.\, [\![\, \mathsf{y} = \mathsf{x}\ ?\ \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathbb{m}^{\sharp}) : \alpha^{\imath}(\{\mathbb{m}(\mathsf{y}) \mid \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp})\})\,]\!]$$

(proof) Cont'd

$$\dot{\alpha}^{\imath} \circ (\lambda \mathsf{y}\,.\,\llbracket\, \mathsf{y} = \mathsf{x} \;?\; \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathbb{m}^{\sharp}) \;\text{⸬}\; \{\mathbb{m}(\mathsf{y}) \mid \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp})\} \,\rrbracket)$$

$=$ by definition of $\dot{\alpha}^{\imath}$

$$\lambda \mathsf{y}\,.\,\llbracket\, \mathsf{y} = \mathsf{x} \;?\; \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathbb{m}^{\sharp}) \;\text{⸬}\; \alpha^{\imath}(\{\mathbb{m}(\mathsf{y}) \mid \mathbb{m} \in \gamma^{\imath}_{nr}(\mathbb{m}^{\sharp})\}) \,\rrbracket$$

$=$ by definition of $\gamma^{\imath}_{nr}$

$$\lambda \mathsf{y}\,.\,\llbracket\, \mathsf{y} = \mathsf{x} \;?\; \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathbb{m}^{\sharp}) \;\text{⸬}\; \alpha^{\imath} \circ \gamma^{\imath}(\mathbb{m}^{\sharp}(\mathsf{y})) \,\rrbracket$$

(proof) Cont'd

$\dot{\alpha}^{\imath} \circ (\lambda y.\ (\!|\ y = x\ ?\ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathrm{m}^\sharp) : \{\mathrm{m}(y) \mid \mathrm{m} \in \gamma^{\imath}_{nr}(\mathrm{m}^\sharp)\}\ |\!)\ )$

$=$ by definition of $\dot{\alpha}^{\imath}$

$\lambda y.\ (\!|\ y = x\ ?\ \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathrm{m}^\sharp) : \alpha^{\imath}(\{\mathrm{m}(y) \mid \mathrm{m} \in \gamma^{\imath}_{nr}(\mathrm{m}^\sharp)\})\ |\!)$

$=$ by definition of $\gamma^{\imath}_{nr}$

$\lambda y.\ (\!|\ y = x\ ?\ \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathrm{m}^\sharp) : \alpha^{\imath} \circ \gamma^{\imath}(\mathrm{m}^\sharp(y))\ |\!)$

$\subseteq^\sharp$ by reductivity of $\alpha^{\imath} \circ \gamma^{\imath}$

$\lambda y.\ (\!|\ y = x\ ?\ \textsc{AbsEval}[exp](\mathrm{m}^\sharp) : \mathrm{m}^\sharp(y)\ |\!)$

(proof) Cont'd

$$\dot{\alpha}^{\imath} \circ (\lambda y\,.\, (\!|\, y = x \mathbin{?} \gamma^{\imath} \circ \textsc{AbsEval}[exp](m^{\sharp}) \mathbin{⁏} \{m(y) \mid m \in \gamma^{\imath}_{nr}(m^{\sharp})\} \,|\!))$$

$=$ by definition of $\dot{\alpha}^{\imath}$

$$\lambda y\,.\, (\!|\, y = x \mathbin{?} \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](m^{\sharp}) \mathbin{⁏} \alpha^{\imath}(\{m(y) \mid m \in \gamma^{\imath}_{nr}(m^{\sharp})\}) \,|\!)$$

$=$ by definition of $\gamma^{\imath}_{nr}$

$$\lambda y\,.\, (\!|\, y = x \mathbin{?} \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](m^{\sharp}) \mathbin{⁏} \alpha^{\imath} \circ \gamma^{\imath}(m^{\sharp}(y)) \,|\!)$$

$\subseteq^{\sharp}$ by reductivity of $\alpha^{\imath} \circ \gamma^{\imath}$

$$\lambda y\,.\, (\!|\, y = x \mathbin{?} \textsc{AbsEval}[exp](m^{\sharp}) \mathbin{⁏} m^{\sharp}(y) \,|\!)$$

$=$

$$m^{\sharp}[x \leftarrow \textsc{AbsEval}[exp](m^{\sharp})]$$

(proof) Cont'd

$$\dot{\alpha}^{\imath} \circ (\lambda \mathtt{y}\,.\, (\!| \mathtt{y} = \mathtt{x} \,?\, \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathfrak{m}^\sharp) \,:\, \{\mathfrak{m}(\mathtt{y}) \mid \mathfrak{m} \in \gamma^{\imath}_{nr}(\mathfrak{m}^\sharp)\} |\!))$$

$=$ by definition of $\dot{\alpha}^{\imath}$

$$\lambda \mathtt{y}\,.\, (\!| \mathtt{y} = \mathtt{x} \,?\, \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathfrak{m}^\sharp) \,:\, \alpha^{\imath}(\{\mathfrak{m}(\mathtt{y}) \mid \mathfrak{m} \in \gamma^{\imath}_{nr}(\mathfrak{m}^\sharp)\}) |\!)$$

$=$ by definition of $\gamma^{\imath}_{nr}$

$$\lambda \mathtt{y}\,.\, (\!| \mathtt{y} = \mathtt{x} \,?\, \alpha^{\imath} \circ \gamma^{\imath} \circ \textsc{AbsEval}[exp](\mathfrak{m}^\sharp) \,:\, \alpha^{\imath} \circ \gamma^{\imath}(\mathfrak{m}^\sharp(\mathtt{y})) |\!)$$

$\sqsubseteq^\sharp$ by reductivity of $\alpha^{\imath} \circ \gamma^{\imath}$

$$\lambda \mathtt{y}\,.\, (\!| \mathtt{y} = \mathtt{x} \,?\, \textsc{AbsEval}[exp](\mathfrak{m}^\sharp) \,:\, \mathfrak{m}^\sharp(\mathtt{y}) |\!)$$

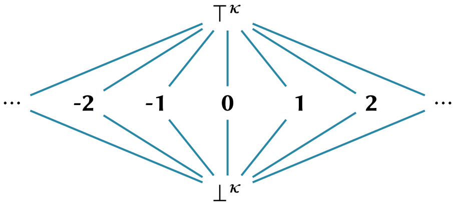
$=$

$$\mathfrak{m}^\sharp[\mathtt{x} \hookleftarrow \textsc{AbsEval}[exp](\mathfrak{m}^\sharp)]$$

$=$ by definition of $\mathbf{S}^\sharp$

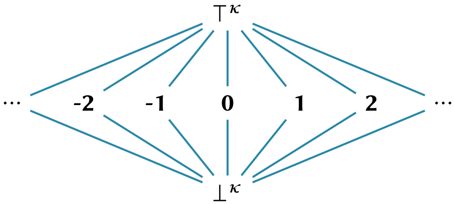
$$\mathbf{S}^\sharp_{\mathtt{x}=exp}(\mathfrak{m}^\sharp)$$

Complete lattice $\langle \mathbb{Z}^{\kappa}, \subseteq^{\kappa}, \cup^{\kappa}, \cap^{\kappa}, \bot^{\kappa}, \top^{\kappa} \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha^{\kappa}}{\overset{\gamma^{\kappa}}{\leftrightarrows}} \langle \mathbb{Z}^{\kappa}, \subseteq^{\kappa} \rangle$



Constants abstract domain

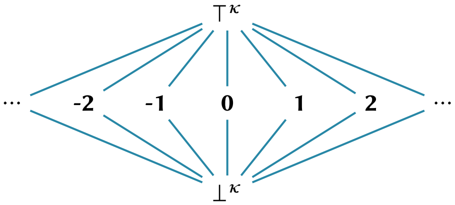
Complete lattice $\langle \mathbb{Z}^\kappa, \subseteq^\kappa, \cup^\kappa, \cap^\kappa, \bot^\kappa, \top^\kappa \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha^\kappa}{\overset{\gamma^\kappa}{\leftrightarrows}} \langle \mathbb{Z}^\kappa, \subseteq^\kappa \rangle$



$\subseteq^\kappa, \cup^\kappa, \cap^\kappa$ trivially defined

Constants abstract domain

Complete lattice $\langle \mathbb{Z}^{\kappa}, \subseteq^{\kappa}, \cup^{\kappa}, \cap^{\kappa}, \perp^{\kappa}, \top^{\kappa} \rangle$ with $\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha^{\kappa}}{\overset{\gamma^{\kappa}}{\leftrightarrows}} \langle \mathbb{Z}^{\kappa}, \subseteq^{\kappa} \rangle$



$\subseteq^{\kappa}, \cup^{\kappa}, \cap^{\kappa}$ trivially defined

Abstraction $\alpha^{\kappa} \triangleq \lambda X . \begin{cases} \perp^{\kappa} & \text{if } X = \emptyset \\ \mathbf{z} & \text{if } X = \{z\} \\ \top^{\kappa} & \text{otherwise} \end{cases}$ and concretization $\gamma^{\kappa} \triangleq \lambda \mathrm{v}^{\kappa} . \begin{cases} \emptyset & \text{if } \mathrm{v}^{\kappa} = \perp^{\kappa} \\ \{z\} & \text{if } \mathrm{v}^{\kappa} = \mathbf{z} \\ \mathbb{Z} & \text{otherwise} \end{cases}$

Abstract arithmetic operations

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\kappa(\{z\}) = \mathbf{z}$$

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\kappa(\{z\}) = \mathbf{z}$$

$$\begin{aligned} \mathbb{v}_1^\kappa +^\sharp \mathbb{v}_2^\kappa &\triangleq \alpha^\kappa(\{z_1 + z_2 \mid z_1 \in \gamma^\kappa(\mathbb{v}_1^\kappa) \land z_2 \in \gamma^\kappa(\mathbb{v}_2^\kappa)\}) \\ &= \begin{cases} \bot^\kappa & \text{if } \mathbb{v}_1^\kappa = \bot^\kappa \lor \mathbb{v}_2^\kappa = \bot^\kappa \\ \top^\kappa & \text{if } \mathbb{v}_1^\kappa = \top^\kappa \lor \mathbb{v}_2^\kappa \in \top^\kappa \\ \mathbf{z} & \text{if } \mathbb{v}_1^\kappa = \mathbf{z}_1 \land \mathbb{v}_2^\kappa = \mathbf{z}_2 \land z = z_1 + z_2 \end{cases} \end{aligned}$$

Abstract arithmetic operations

$$z^\sharp \triangleq \alpha^\kappa(\{z\}) = \mathbf{z}$$

$$\mathbb{v}_1^\kappa +^\sharp \mathbb{v}_2^\kappa \triangleq \alpha^\kappa(\{z_1 + z_2 \mid z_1 \in \gamma^\kappa(\mathbb{v}_1^\kappa) \wedge z_2 \in \gamma^\kappa(\mathbb{v}_2^\kappa)\})$$
$$= \begin{cases} \bot^\kappa & \text{if } \mathbb{v}_1^\kappa = \bot^\kappa \vee \mathbb{v}_2^\kappa = \bot^\kappa \\ \top^\kappa & \text{if } \mathbb{v}_1^\kappa = \top^\kappa \vee \mathbb{v}_2^\kappa \in \top^\kappa \\ \mathbf{z} & \text{if } \mathbb{v}_1^\kappa = \mathbf{z}_1 \wedge \mathbb{v}_2^\kappa = \mathbf{z}_2 \wedge z = z_1 + z_2 \end{cases}$$

$$\mathbb{v}_1^\kappa \times^\sharp \mathbb{v}_2^\kappa \triangleq \alpha^\kappa(\{z_1 \cdot z_2 \mid z_1 \in \gamma^\kappa(\mathbb{v}_1^\kappa) \wedge z_2 \in \gamma^\kappa(\mathbb{v}_2^\kappa)\})$$
$$= \begin{cases} \bot^\kappa & \text{if } \mathbb{v}_1^\kappa = \bot^\kappa \vee \mathbb{v}_2^\kappa = \bot^\kappa \\ \mathbf{0} & \text{if } \mathbb{v}_1^\kappa = \mathbf{0} \vee \mathbb{v}_2^\kappa = \mathbf{0} \\ \top^\kappa & \text{if } \mathbb{v}_1^\kappa = \top^\kappa \vee \mathbb{v}_2^\kappa \in \top^\kappa \\ \mathbf{z} & \text{if } \mathbb{v}_1^\kappa = \mathbf{z}_1 \wedge \mathbb{v}_2^\kappa = \mathbf{z}_2 \wedge z = z_1 \cdot z_2 \end{cases}$$

Abstract conditions/boolean expressions for $\mathbb{Z}^\kappa$

Abstract conditions/boolean expressions for $\mathbb{Z}^{\kappa}$

$$(\!|\,{}^{\ell}\mathtt{x} == z\,|\!)^{\sharp}\mathfrak{m}^{\sharp} \triangleq \mathbf{S}^{\sharp}_{\mathtt{x}==z}(\mathfrak{m}^{\sharp}) \triangleq \begin{cases} \mathfrak{m}^{\sharp}_{\perp} & \text{if } \mathfrak{m}^{\sharp}(\mathtt{x}) \notin \{\mathbf{z}, \top^{\kappa}\} \\ \mathfrak{m}^{\sharp}[\mathtt{x} \hookleftarrow \mathbf{z}] & \text{otherwise} \end{cases}$$

Abstract conditions/boolean expressions for $\mathbb{Z}^\kappa$

$$(\!|\, {}^{\ell}\mathtt{x} == z \,|\!)^\sharp \mathfrak{m}^\sharp \triangleq \mathbf{S}^\sharp_{\mathtt{x}==z}(\mathfrak{m}^\sharp) \triangleq \begin{cases} \mathfrak{m}^\sharp_\bot & \text{if } \mathfrak{m}^\sharp(\mathtt{x}) \notin \{\mathbf{z}, \top^\kappa\} \\ \mathfrak{m}^\sharp[\mathtt{x} \hookleftarrow \mathbf{z}] & \text{otherwise} \end{cases}$$

$$(\!|\, {}^{\ell}\mathtt{x} == \mathtt{y} + z \,|\!)^\sharp \mathfrak{m}^\sharp \triangleq \mathbf{S}^\sharp_{\mathtt{x}==\mathtt{y}+z}(\mathfrak{m}^\sharp) \triangleq$$

$$\left(\begin{cases} (\!|\, {}^{\ell}\mathtt{x} == w + z \,|\!)^\sharp \mathfrak{m}^\sharp = \mathbf{S}^\sharp_{\mathtt{x}==w+z}(\mathfrak{m}^\sharp) & \text{if } \mathfrak{m}^\sharp(\mathtt{y}) = \mathbf{w} \\ \mathfrak{m}^\sharp & \text{otherwise} \end{cases} \right)$$

$$\cap^\kappa$$

$$\left(\begin{cases} (\!|\, {}^{\ell}\mathtt{y} == w - z \,|\!)^\sharp \mathfrak{m}^\sharp = \mathbf{S}^\sharp_{\mathtt{y}==w-z}(\mathfrak{m}^\sharp) & \text{if } \mathfrak{m}^\sharp(\mathtt{x}) = \mathbf{w} \\ \mathfrak{m}^\sharp & \text{otherwise} \end{cases} \right)$$

Constants analysis: example

$\mathbb{Z}^\kappa$ has finite height: convergence guaranteed in finite time

Constants analysis: example

$\mathbb{Z}^{\kappa}$ has finite height: convergence guaranteed in finite time

```
1 x = 0;
2 y = 10;
3 while (x < 100) {
    4 y = y - 3;
    5 x = x + y;
    6 y = y + 3
7 }
```

$\mathbb{Z}^\kappa$ has finite height: convergence guaranteed in finite time

```
1 x = 0;
2 y = 10;
3 while (x < 100) {
    4 y = y - 3;
    5 x = x + y;
    6 y = y + 3
7 }
```

The constants analysis at 6 finds: $x = \top^\kappa \wedge y = \mathbf{7}$

even if $x \equiv_7 0$

Constants analysis: example

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

$\mathbb{Z}^{\kappa}$ has finite height: convergence guaranteed in finite time

```
1 x = 0;
2 y = 10;
3 while (x < 100) {
    4 y = y - 3;
    5 x = x + y;
    6 y = y + 3
7 }
```

refactoring $\longrightarrow$

```
1 x = 0;
2 while (x < 100) {
    3 x = x + 7;
4 }
```

The constants analysis at 6 finds: $x = \top^{\kappa} \wedge y = \mathbf{7}$

even if $x \equiv_7 0$

Constants analysis: example

$\mathbb{Z}^\kappa$ has finite height: convergence guaranteed in finite time

```
1 x = 0;
2 y = 10;
3 while (x < 100) {
    4 y = y - 3;
    5 x = x + y;
    6 y = y + 3
7 }
```

refactoring $\longrightarrow$

```
1 x = 0;
2 while (x < 100) {
    3 x = x + 7;
4 }
```

The constants analysis at 6 finds: $x = \top^\kappa \wedge y = \mathbf{7}$ even if $x \equiv_7 0$

The analysis can find constants that do not appear syntactically in the program!

Non-relational Extrapolation

$\mathbb{Z}^\imath$ has infinite height, so does $\mathbb{M}^\sharp$

- Abstract computation may not converge in finite time
- We need an extrapolation operator (widening) to force termination

$\mathbb{Z}^\imath$ has infinite height, so does $\mathbb{M}^\sharp$

- Abstract computation may not converge in finite time
- We need an extrapolation operator (widening) to force termination

$$\text{Widening operator } \nabla : \mathbb{M}^\sharp \times \mathbb{M}^\sharp \to \mathbb{M}^\sharp$$

$\mathbb{Z}^\imath$ has infinite height, so does $\mathbb{M}^\sharp$

- Abstract computation may not converge in finite time
- We need an extrapolation operator (widening) to force termination

$$\text{Widening operator } \nabla : \mathbb{M}^\sharp \times \mathbb{M}^\sharp \to \mathbb{M}^\sharp$$

Soundness: $\gamma^\imath_{nr}(\mathfrak{m}^\sharp_1) \cup \gamma^\imath_{nr}(\mathfrak{m}^\sharp_2) \subseteq \gamma^\imath_{nr}(\mathfrak{m}^\sharp_1 \;\nabla\; \mathfrak{m}^\sharp_2)$

$\mathbb{Z}^\imath$ has infinite height, so does $\mathbb{M}^\sharp$

- Abstract computation may not converge in finite time
- We need an extrapolation operator (widening) to force termination

Widening operator $\nabla : \mathbb{M}^\sharp \times \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

Soundness: $\gamma^\imath_{nr}(\mathfrak{m}^\sharp_1) \cup \gamma^\imath_{nr}(\mathfrak{m}^\sharp_2) \subseteq \gamma^\imath_{nr}(\mathfrak{m}^\sharp_1 \;\nabla\; \mathfrak{m}^\sharp_2)$

Termination: for every chain $\left(\mathfrak{m}^\sharp_i\right)_{i>0}$, the increasing chain $\left(\overline{\mathfrak{m}}^\sharp_i\right)_{i>0}$ defined as

$$\begin{cases} \overline{\mathfrak{m}}^\sharp_0 & \triangleq \quad \mathfrak{m}^\sharp_0 \\ \overline{\mathfrak{m}}^\sharp_{n+1} & \triangleq \quad \overline{\mathfrak{m}}^\sharp_n \;\nabla\; \mathfrak{m}^\sharp_{n+1} \end{cases}$$

is stationary, i.e., there exists $k \in \mathbb{N}$ such that $\overline{\mathfrak{m}}^\sharp_{k+1} = \overline{\mathfrak{m}}^\sharp_k$

Intervals widening $\nabla^{\imath} : \mathbb{Z}^{\imath} \times \mathbb{Z}^{\imath} \to \mathbb{Z}^{\imath}$

Intervals widening $\nabla^\imath : \mathbb{Z}^\imath \times \mathbb{Z}^\imath \to \mathbb{Z}^\imath$

unstable bounds set to infinity

$$\bot^\imath \ \nabla^\imath \ [a, b] \triangleq [a, b] \ \nabla^\imath \ \bot^\imath \triangleq [a, b]$$

$$[a, a'] \ \nabla^\imath \ [b, b'] \triangleq [(\!| b < a \ ? \ \text{-}\infty \ ⁏ \ a |\!), (\!| b' > a' \ ? \ \infty \ ⁏ \ a' |\!)]$$

Intervals widening $\nabla^\imath : \mathbb{Z}^\imath \times \mathbb{Z}^\imath \to \mathbb{Z}^\imath$ unstable bounds set to infinity

$$\bot^\imath \ \nabla^\imath \ [a,b] \triangleq [a,b] \ \nabla^\imath \ \bot^\imath \triangleq [a,b]$$
$$[a,a'] \ \nabla^\imath \ [b,b'] \triangleq [(\!| b < a \ ? \ \text{-}\infty \ : a |\!), (\!| b' > a' \ ? \ \infty \ : a' |\!)]$$

Point-wise lift of intervals widening to (abstract) memories $\nabla : \mathbb{M}^\sharp \times \mathbb{M}^\sharp \to \mathbb{M}^\sharp$

$$\mathtt{m}^\sharp_1 \ \nabla \ \mathtt{m}^\sharp_2 \triangleq \lambda \mathtt{x} \, . \, \mathtt{m}^\sharp_1(\mathtt{x}) \ \nabla^\imath \ \mathtt{m}^\sharp_2(\mathtt{x})$$

Intervals widening $\nabla^{\imath} : \mathbb{Z}^{\imath} \times \mathbb{Z}^{\imath} \to \mathbb{Z}^{\imath}$ unstable bounds set to infinity

$$\bot^{\imath}\ \nabla^{\imath}\ [a,b] \triangleq [a,b]\ \nabla^{\imath}\ \bot^{\imath} \triangleq [a,b]$$
$$[a,a']\ \nabla^{\imath}\ [b,b'] \triangleq [(\!|\, b < a \;?\; \text{-}\infty \;:\; a \,|\!), (\!|\, b' > a' \;?\; \infty \;:\; a' \,|\!)]$$

Point-wise lift of intervals widening to (abstract) memories $\nabla : \mathbb{M}^{\sharp} \times \mathbb{M}^{\sharp} \to \mathbb{M}^{\sharp}$

$$\mathtt{m}^{\sharp}_1\ \nabla\ \mathtt{m}^{\sharp}_2 \triangleq \lambda \mathtt{x}\,.\ \mathtt{m}^{\sharp}_1(\mathtt{x})\ \nabla^{\imath}\ \mathtt{m}^{\sharp}_2(\mathtt{x})$$

The construction works for generic non-relational abstractions

Define a set $\mathscr{W}$ of widening points such that every CFG cycle has a point in $\mathscr{W}$

Define a set $\mathscr{W}$ of widening points such that every CFG cycle has a point in $\mathscr{W}$

The abstract reachability semantics with widening is the least solution of the system of equations

$$\begin{cases} X_{0} \triangleq \mathbb{m}^{\sharp}_{\top} & \\ X_{\ell} \triangleq \bigcup^{\sharp}_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt |\!)^{\sharp} X_{\ell'} & \text{if } \ell \notin \mathscr{W} \land \text{next}\left({}^{\ell'} stmt\right) = \ell \\ X_{\ell} \triangleq X_{\ell} \mathbin{\nabla} \bigcup^{\sharp}_{\ell' \in \mathbb{L}} (\!| {}^{\ell'} stmt |\!)^{\sharp} X_{\ell'} & \text{if } \ell \in \mathscr{W} \land \text{next}\left({}^{\ell'} stmt\right) = \ell \end{cases}$$

Define a set $\mathscr{W}$ of widening points such that every CFG cycle has a point in $\mathscr{W}$

The abstract reachability semantics with widening is the least solution of the system of equations

$$\begin{cases} X_{0} \triangleq \mathbb{m}^{\sharp}_{\top} & \\ X_{\ell} \triangleq \bigcup^{\sharp}_{\ell' \in \mathbb{L}} (\!| \,^{\ell'}stmt\, |\!)^{\sharp} X_{\ell'} & \text{if } \ell \notin \mathscr{W} \wedge \text{next}\left(^{\ell'}stmt\right) = \ell \\ X_{\ell} \triangleq X_{\ell} \mathbin{\nabla} \bigcup^{\sharp}_{\ell' \in \mathbb{L}} (\!| \,^{\ell'}stmt\, |\!)^{\sharp} X_{\ell'} & \text{if } \ell \in \mathscr{W} \wedge \text{next}\left(^{\ell'}stmt\right) = \ell \end{cases}$$

The solution can be computed by increasing iterations

Define a set $\mathscr{W}$ of widening points such that every CFG cycle has a point in $\mathscr{W}$

The abstract reachability semantics with widening is the least solution of the system of equations

$$\begin{cases} X_{\mathtt{0}} \triangleq \mathfrak{m}^\sharp_\top & \\ X_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!)^\sharp X_{\ell'} & \text{if } \ell \notin \mathscr{W} \land \text{next}\left({}^{\ell'}stmt\right) = \ell \\ X_\ell \triangleq X_\ell \, \nabla \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!)^\sharp X_{\ell'} & \text{if } \ell \in \mathscr{W} \land \text{next}\left({}^{\ell'}stmt\right) = \ell \end{cases}$$

The solution can be computed by increasing iterations

$$\begin{cases} X^0_{\mathtt{0}} \triangleq \mathfrak{m}^\sharp_\top \\ X^0_\ell \triangleq \mathfrak{m}^\sharp_\bot \end{cases} \qquad \begin{cases} X^{n+1}_{\mathtt{0}} \triangleq \mathfrak{m}^\sharp_\top & \\ X^{n+1}_\ell \triangleq \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!)^\sharp X^n_{\ell'} & \text{if } \ell \notin \mathscr{W} \\ X^{n+1}_\ell \triangleq X^n_\ell \, \nabla \bigcup^\sharp_{\ell' \in \mathbb{L}} (\!| {}^{\ell'}stmt |\!)^\sharp X^n_{\ell'} & \text{if } \ell \in \mathscr{W} \end{cases}$$

Widening can lead to coarse approximations

Widening can lead to coarse approximations

- Introduce an interpolation operator (narrowing) not always improving precision

Widening can lead to coarse approximations

- Introduce an interpolation operator (narrowing) not always improving precision
- Refine the extrapolation operator

Widening can lead to coarse approximations

- Introduce an interpolation operator (narrowing) not always improving precision
- Refine the extrapolation operator

Define a set T of thresholds (containing $-\infty$ and ∞)

$$[a,a'] \,\nabla^{\imath}_{T}\, [b,b'] \triangleq [\llparenthesis\, b < a \;?\; \curlywedge\{t \in T \mid t \leq b\} : a \,\rrparenthesis, \llparenthesis\, b' > a' \;?\; \curlyvee\{t \in T \mid b' \leq t\} : a' \,\rrparenthesis]$$

Widening with thresholds

Widening can lead to coarse approximations

- Introduce an interpolation operator (narrowing) — not always improving precision
- Refine the extrapolation operator

Define a set T of thresholds (containing $-\infty$ and ∞)

$$[a, a'] \; \nabla_T^{\imath} \; [b, b'] \triangleq [\llparenthesis\, b < a \; ? \; \bigwedge\{t \in T \mid t \leq b\} \; : \; a \,\rrparenthesis, \llparenthesis\, b' > a' \; ? \; \bigvee\{t \in T \mid b' \leq t\} \; : \; a' \,\rrparenthesis]$$

The widening stops at the closest stable bound (not necessarily infinite)

Widening can lead to coarse approximations

- Introduce an interpolation operator (narrowing) not always improving precision
- Refine the extrapolation operator

Define a set T of thresholds (containing -∞ and ∞)

$$[a, a'] \nabla_T^\imath [b, b'] \triangleq [\llparenthesis\, b < a \,?\, \bigwedge\{t \in T \mid t \leq b\} \,:\, a \,\rrparenthesis, \llparenthesis\, b' > a' \,?\, \bigvee\{t \in T \mid b' \leq t\} \,:\, a' \,\rrparenthesis]$$

The widening stops at the closest stable bound (not necessarily infinite)

- Useful when it is easy to find a good T e.g., array bound checking

Widening can lead to coarse approximations

- Introduce an interpolation operator (narrowing) — not always improving precision
- Refine the extrapolation operator

Define a set T of thresholds (containing $-\infty$ and ∞)

$$[a, a'] \nabla^{\imath}_{T} [b, b'] \triangleq [\llbracket b < a \,?\, \bigwedge\{t \in T \mid t \leq b\} \,:\, a \rrbracket, \llbracket b' > a' \,?\, \bigvee\{t \in T \mid b' \leq t\} \,:\, a' \rrbracket]$$

The widening stops at the closest stable bound (not necessarily infinite)

- Useful when it is easy to find a good T — e.g., array bound checking
- Useful when bound over-approximation suffices — e.g., arithmetic overflow checking

Thanks for the attention!

LaTeX is the way