Termination (Resilience) Analysis, and Bugs in Its Implementation

Dagstuhl Seminar 25242 “Testing Program Analyzers and Verifiers”

Caterina Urban (joint work with **Naïm Moussaoui Remil**)
Inria & École Normale Supérieure | Université PSL

Which Non-Termination Alarm is Worse?

```
function f(x) {

  1...
  2z ← 10
  3if ( ... ) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```

diverges when x = 0

diverges when c ≥ 0

non-deterministic value choice

Which Non-Termination Alarm is Worse?

Robust Non-Termination

```
function f(x) {

  1...
  2z ← 10
  3if ( ... ) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```

diverges when x = 0

diverges when c ≥ 0

non-deterministic value choice

Robust Non-Termination

∃ Input ∀ Non-Deterministic Choices : Program Diverges

```
function f(x){                                  demonic non-determinism
  1...
  2z ← 10
  3if ( ... ) then
      while 4(z ≥ 0) do                         🚨 ← diverges when x = 0
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```

Termination Resilience

∀ Inputs ∃ Non-Deterministic Choice : Program Terminates

```
function f(x) {

  1...
  2z ← 10
  3if ( ... ) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```

👌 ⟵ terminates when **c < 0, independently of the value of x**

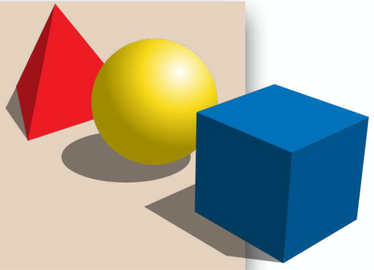
angelic non-determinism

Termination Resilience Static Analysis

3-Step Recipe

practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties



concrete semantics
mathematical models of the program behavior

Termination Resilience Static Analysis

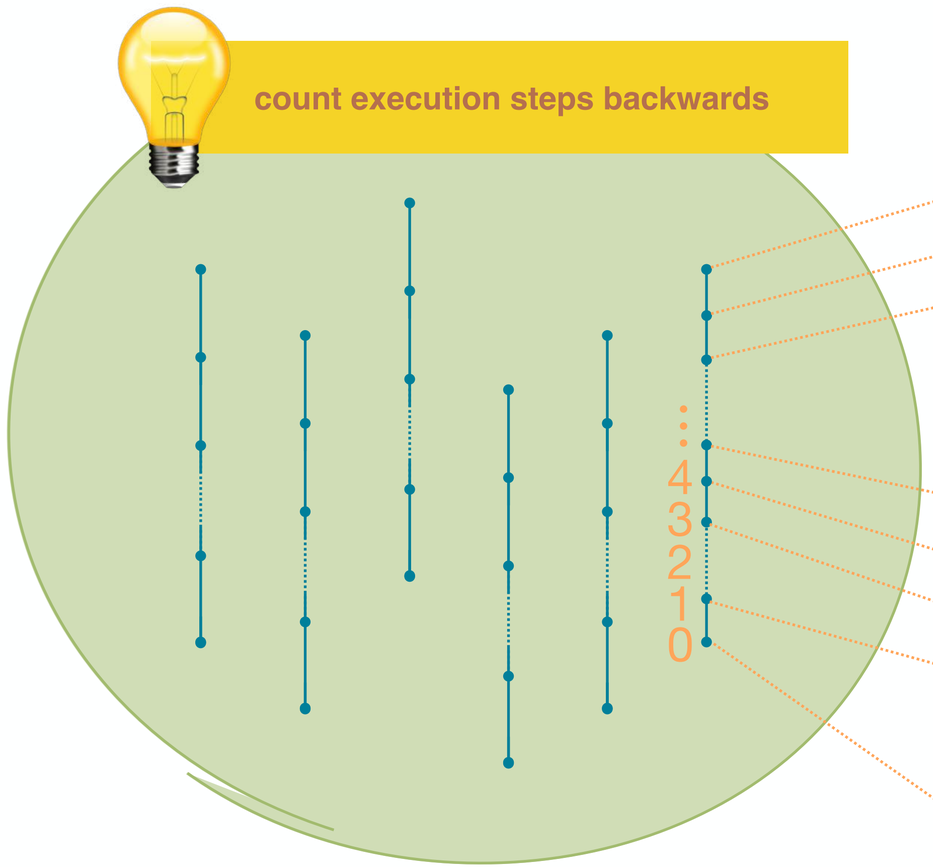
3-Step Recipe

practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties

concrete semantics
mathematical models of the program behavior

Termination Resilience Semantics




```
function f(x) {

  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```

Termination Resilience Semantics

$$f_1 \sqsubseteq f_2 \overset{\text{def}}{=} \mathrm{dom}(f_1) \subseteq \mathrm{dom}(f_2) \wedge \forall x \in \mathrm{dom}(f_1) : f_1(x) \leq f_2(x)$$

$$\Theta \overset{\text{def}}{=} \mathrm{lfp}^{\sqsubseteq}_{\dot{\varnothing}}\ \lambda f\, \lambda s\,.\ \begin{cases} 0 & \text{final states} \quad s \in \Omega_\tau \\ \sup\{f(s') + 1 \mid \langle s, s' \rangle \in \tau\} & s \in \widetilde{\mathrm{pre}}_{\tau^{\mathsf{i}}}(\mathrm{dom}(f)) \\ \inf\{f(s') + 1 \mid \langle s, s' \rangle \in \tau\} & s \in \mathrm{pre}_{\tau^{\mathsf{r}}}(\mathrm{dom}(f)) \\ \text{undefined} & \text{otherwise} \end{cases}$$

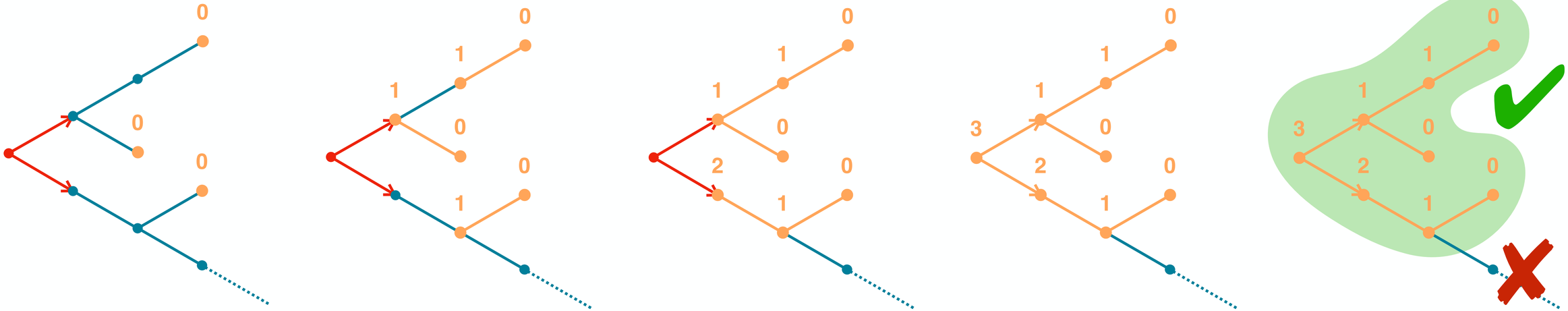
$$\widetilde{\mathrm{pre}}_{\tau^{\mathsf{i}}}(X) \overset{\text{def}}{=} \{s \mid \forall s' \colon \langle s, s' \rangle \in \tau^{\mathsf{i}} \Rightarrow s' \in X\}$$

input transitions

regular transitions

$$\mathrm{pre}_{\tau^{\mathsf{r}}}(X) \overset{\text{def}}{=} \{s \mid \exists s' \in X \colon \langle s, s' \rangle \in \tau^{\mathsf{r}}\}$$

totally undefined function

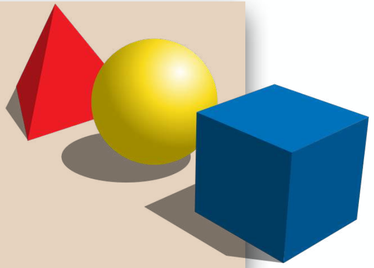


Termination Resilience Static Analysis

3-Step Recipe

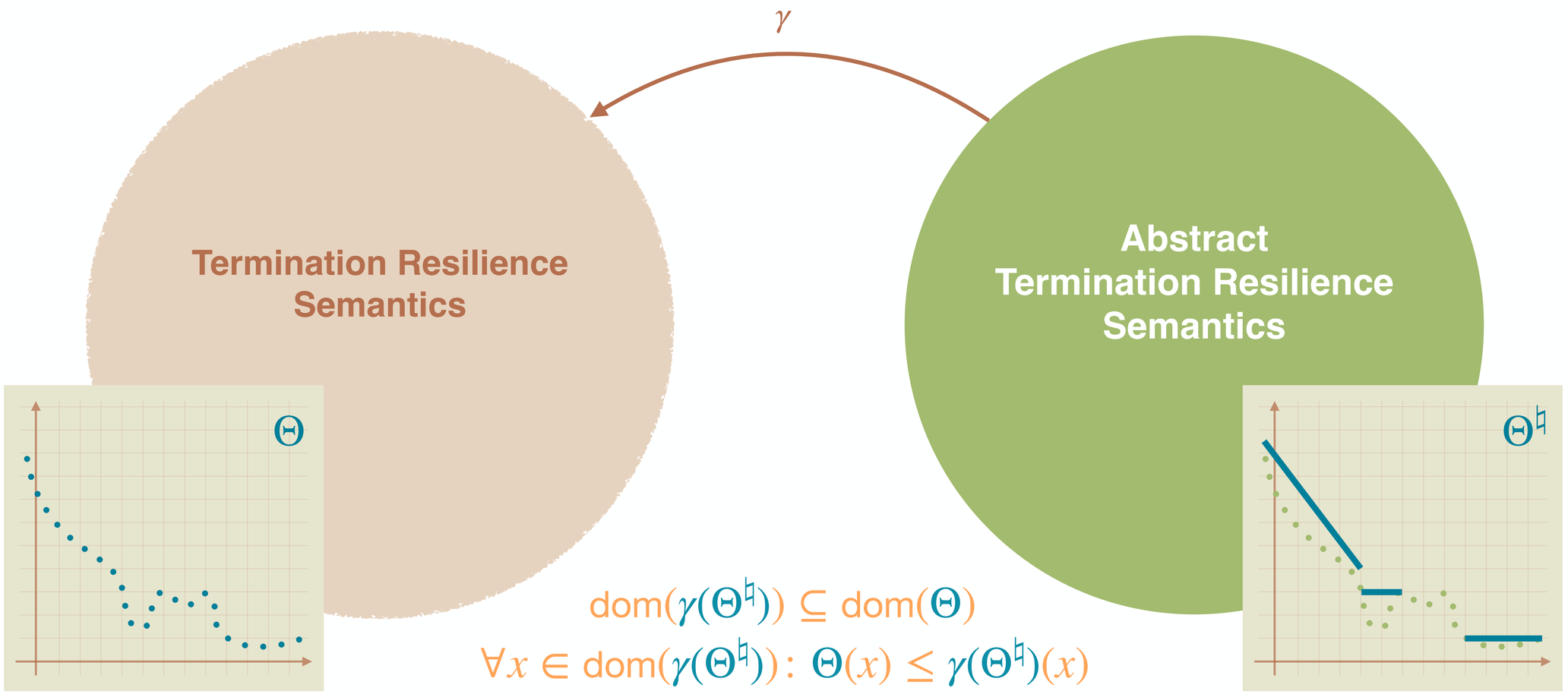
practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties

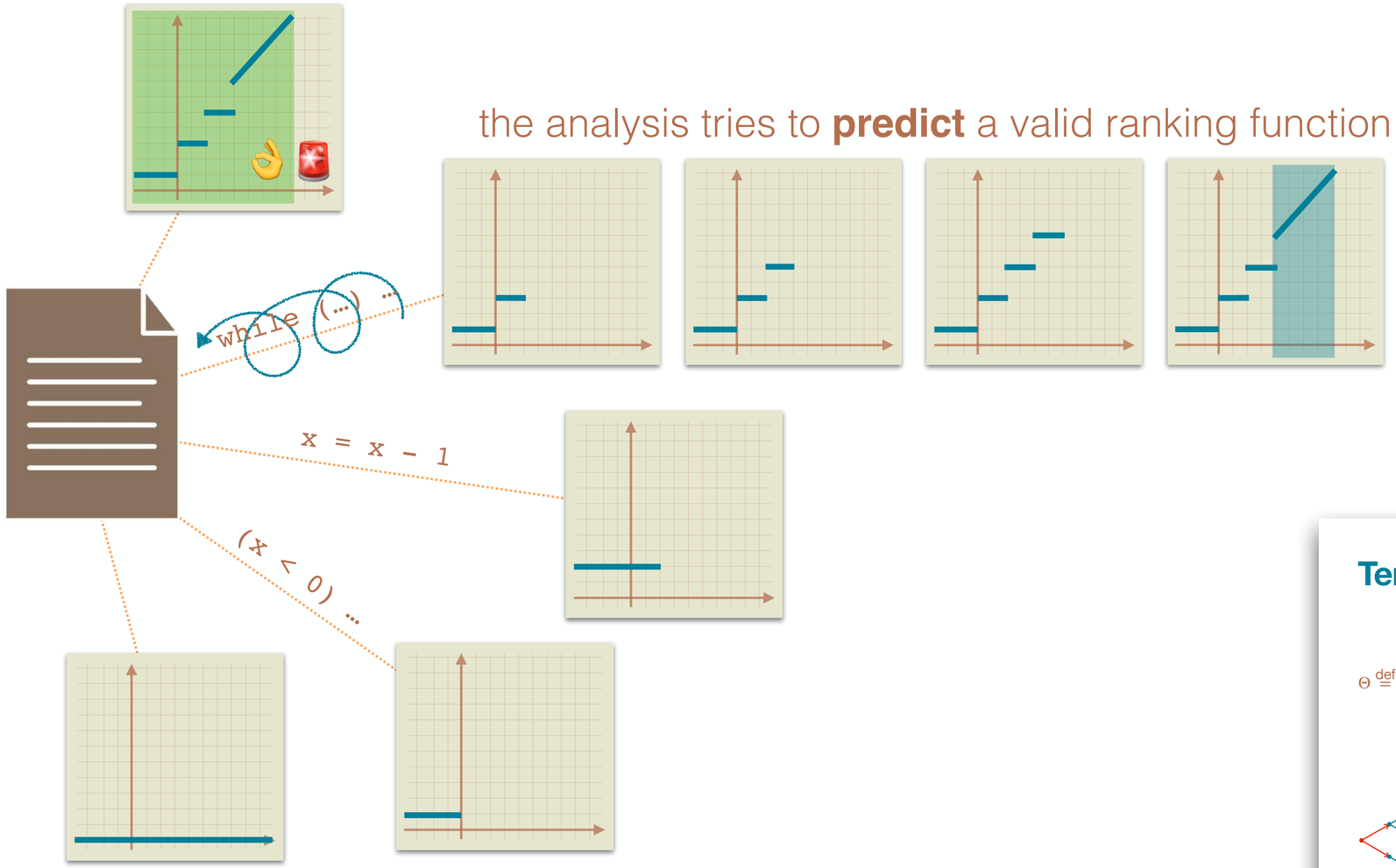


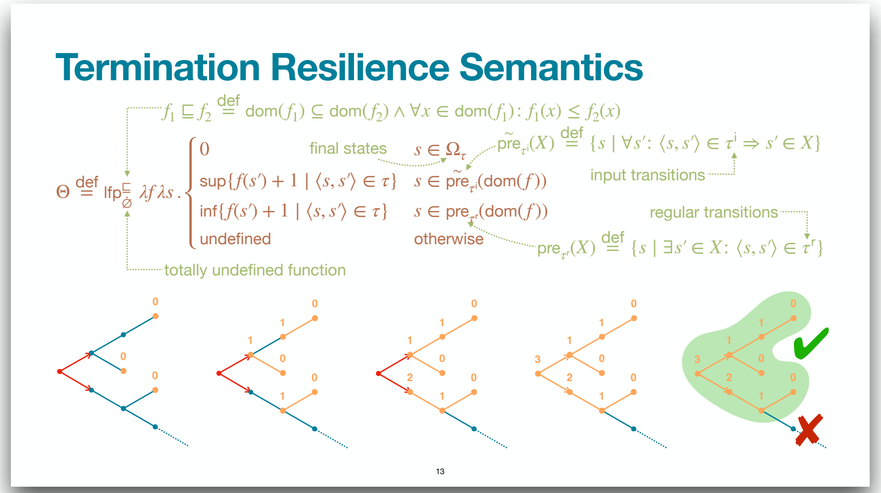
concrete semantics
mathematical models of the program behavior

Piecewise-Defined Ranking Functions

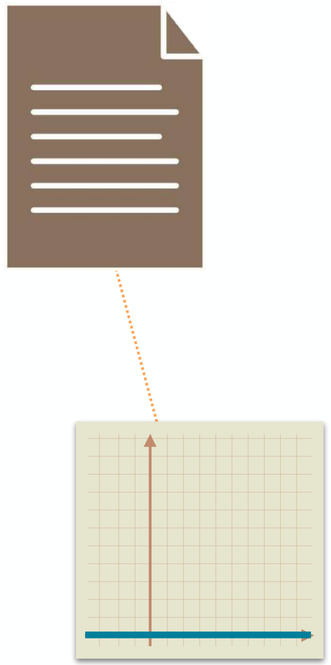


Termination Resilience Static Analysis





Termination Resilience Static Analysis

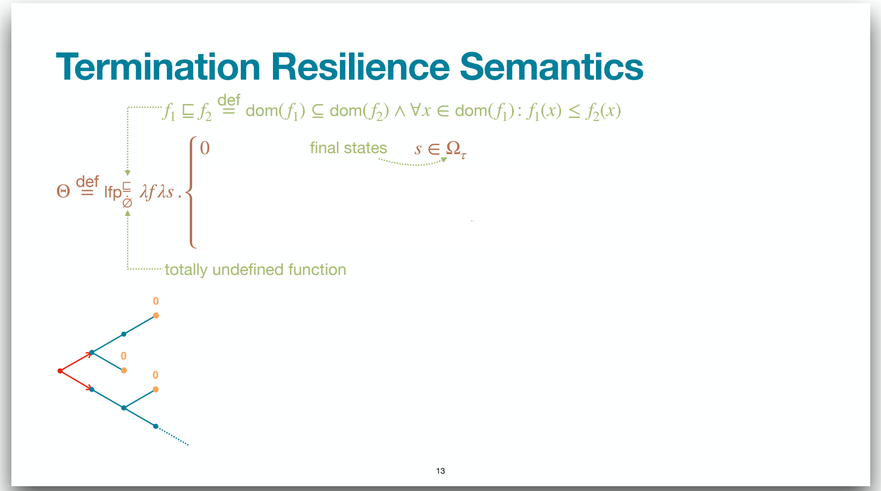


Termination Resilience Static Analysis

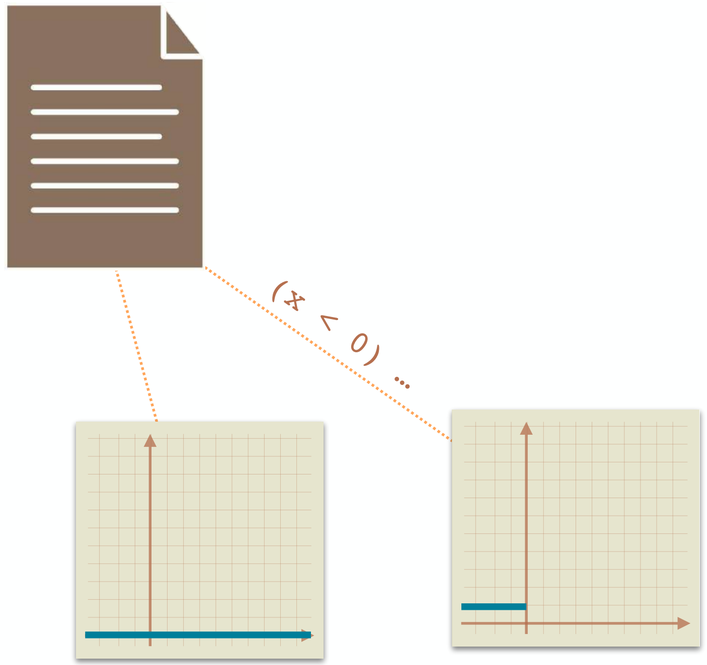
Static Backward Analysis

```
function f(x) {

  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```

λxzac. 0



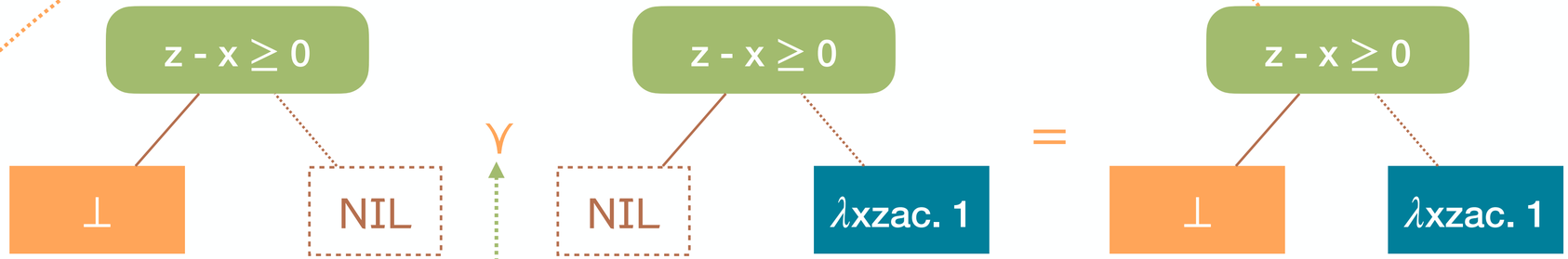
Termination Resilience Static Analysis



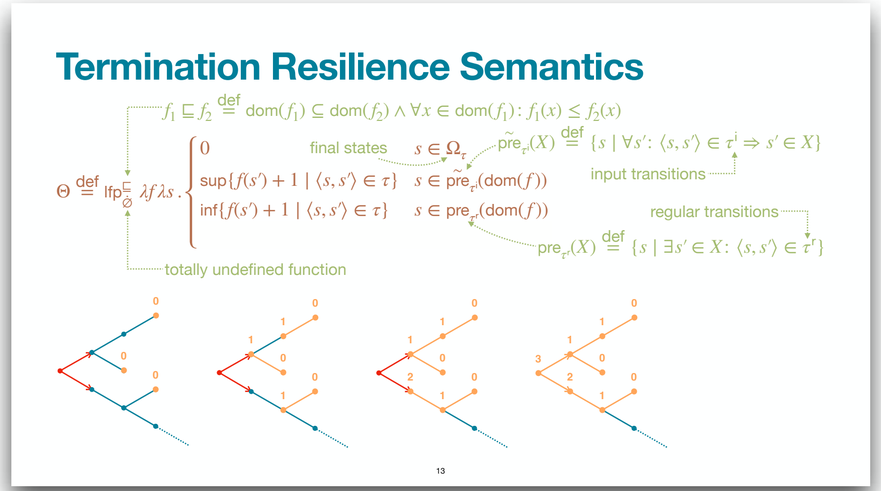
Termination Resilience Static Analysis

Boolean Conditions

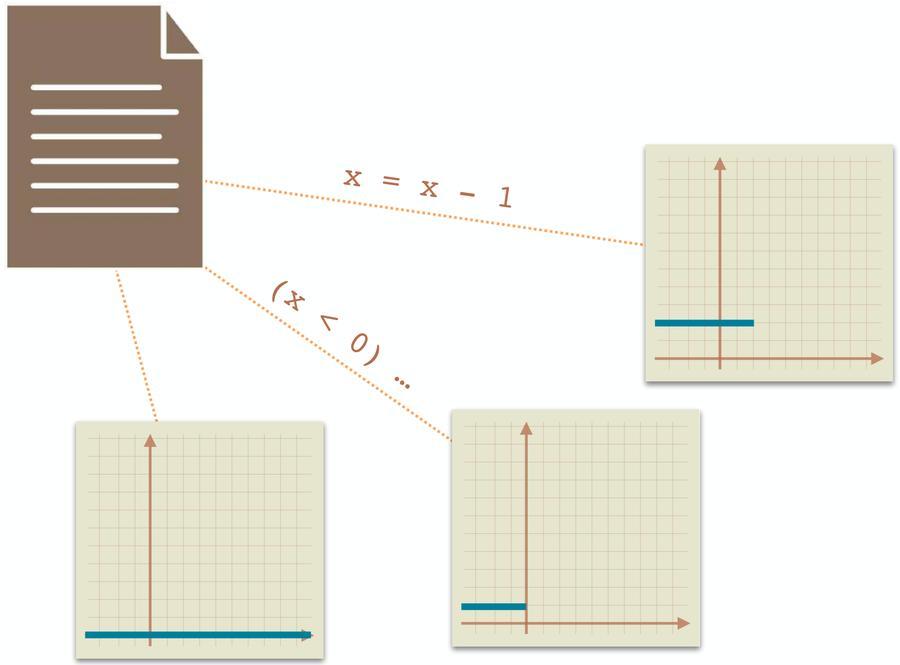
```
function f(x) {
  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi
}11
```



λxzac. 0



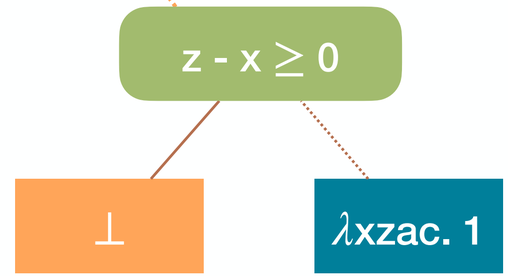
Termination Resilience Static Analysis

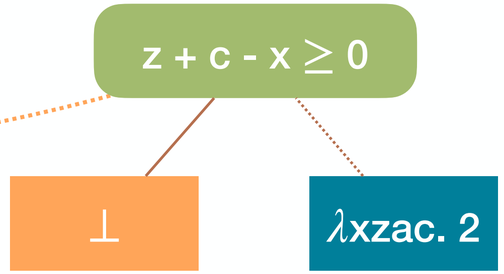


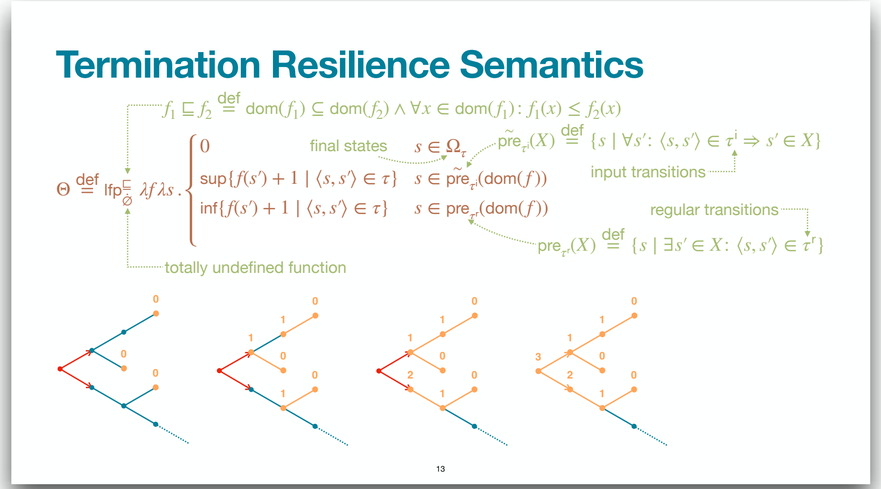
Termination Resilience Static Analysis

Variable Assignment

```
function f(x) {

  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```



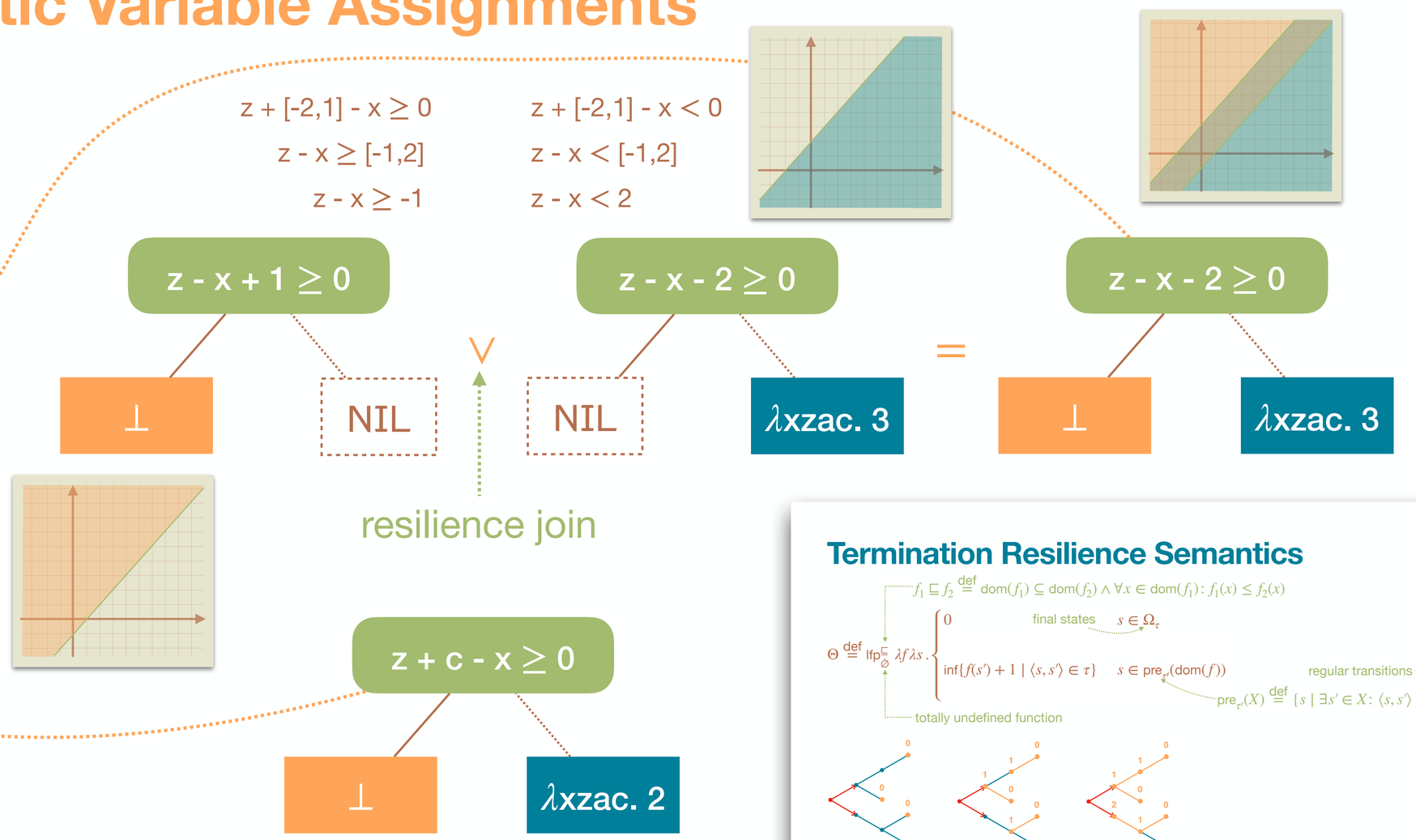




Termination Resilience Static Analysis

Non-Deterministic Variable Assignments

```
function f(x) {

  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```



Termination Resilience Semantics

$$f_1 \sqsubseteq f_2 \stackrel{\text{def}}{=} \text{dom}(f_1) \subseteq \text{dom}(f_2) \wedge \forall x \in \text{dom}(f_1) : f_1(x) \leq f_2(x)$$

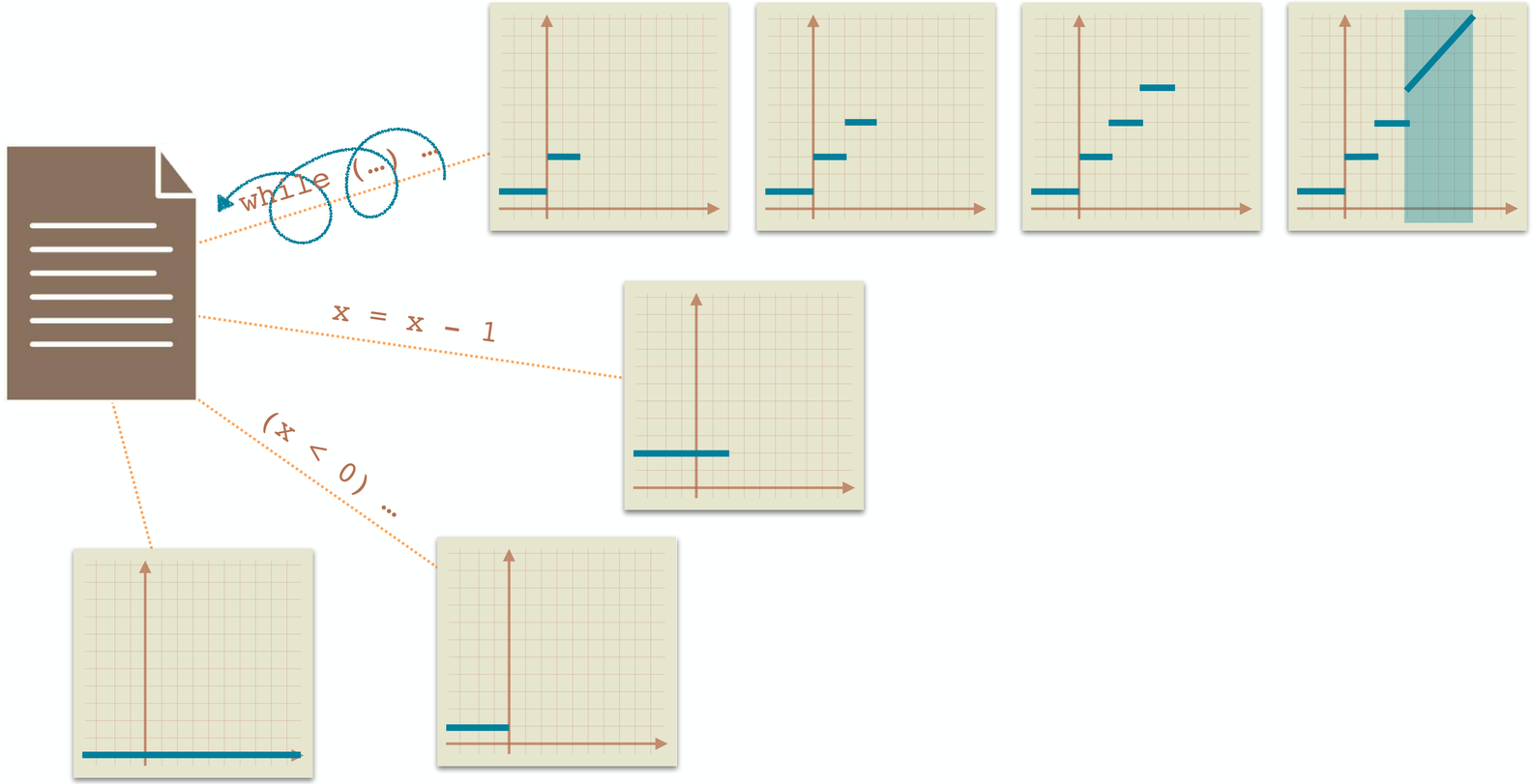
$$\Theta \stackrel{\text{def}}{=} \text{lfp}^{\sqsubseteq}_{\varnothing}\ \lambda f \lambda s .\ \begin{cases} 0 & s \in \Omega_\tau \quad \text{(final states)} \\ \inf\{f(s') + 1 \mid \langle s, s' \rangle \in \tau\} & s \in \text{pre}_\tau(\text{dom}(f)) \quad \text{(regular transitions)} \end{cases}$$

$$\text{pre}_\tau(X) \stackrel{\text{def}}{=} \{s \mid \exists s' \in X : \langle s, s' \rangle \in \tau\}$$

totally undefined function

Termination Resilience Static Analysis

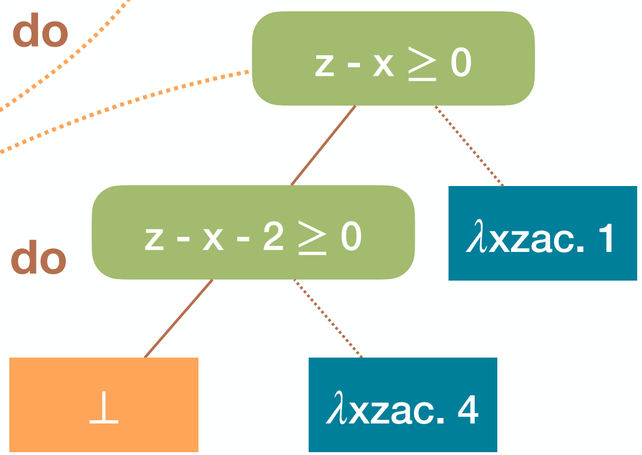
the analysis tries to **predict** a valid ranking function

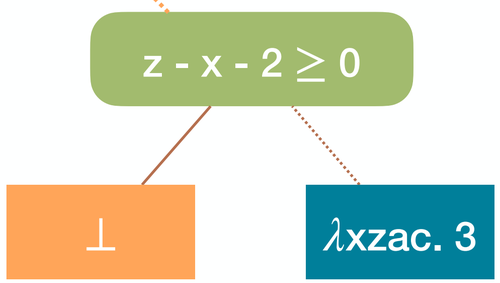


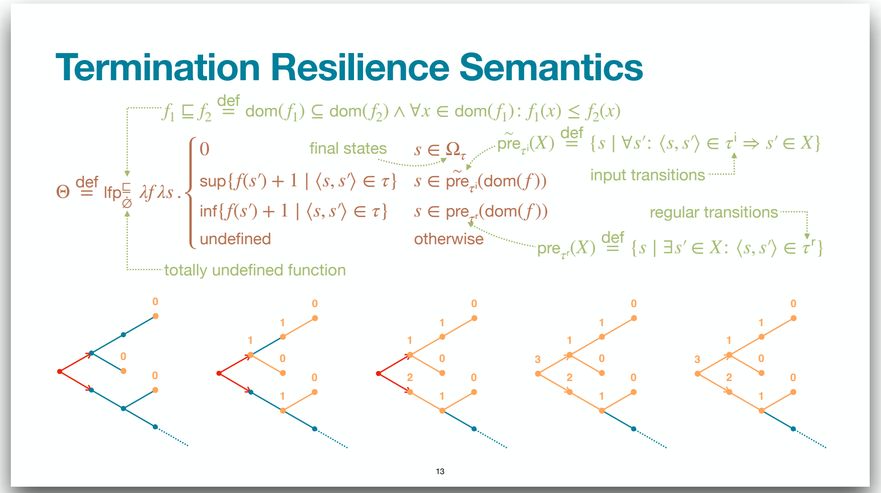
Termination Resilience Static Analysis

Loops

```
function f(x) {

  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```



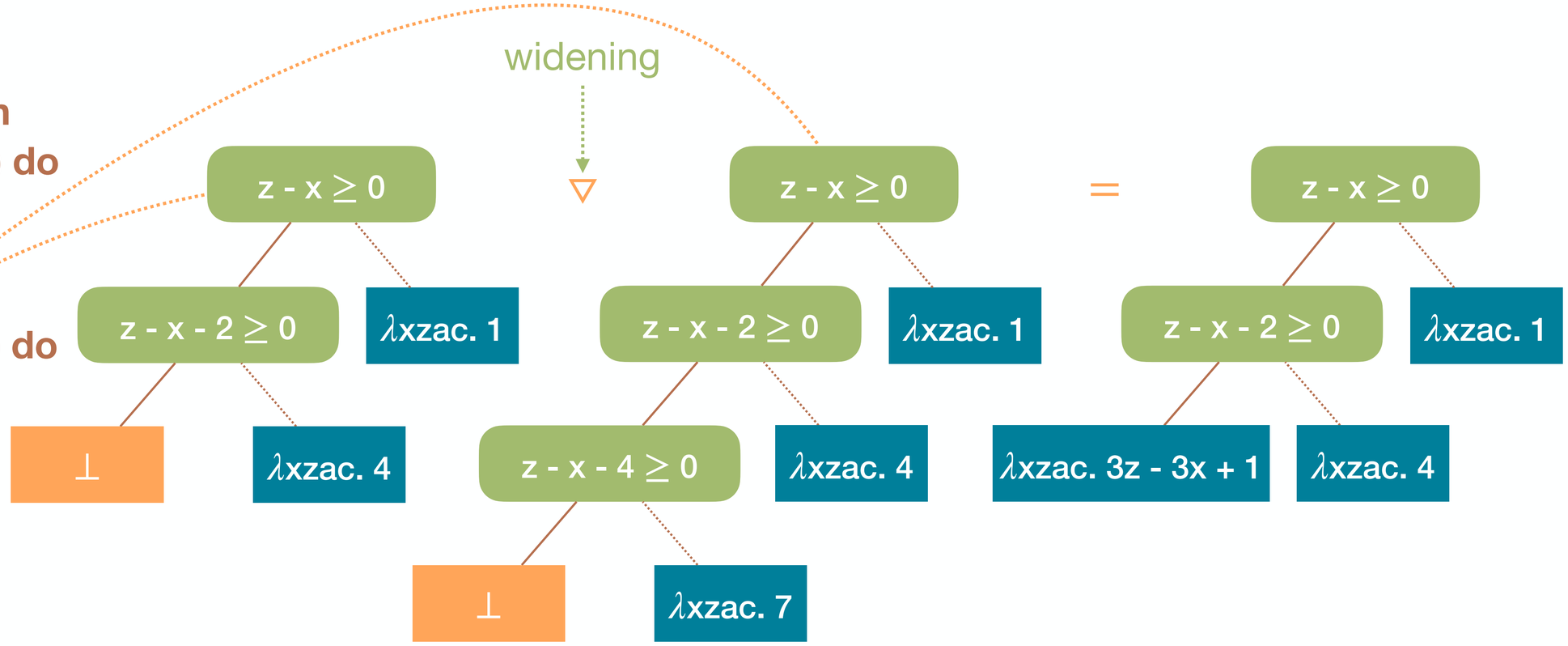




Termination Resilience Static Analysis

Loops

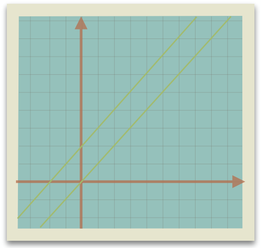
```
function f(x) {
  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi
}11
```

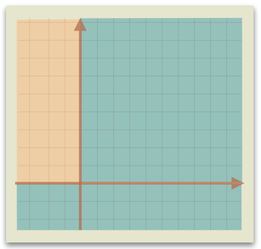


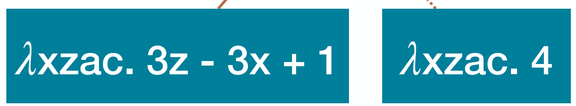
Termination Resilience Static Analysis

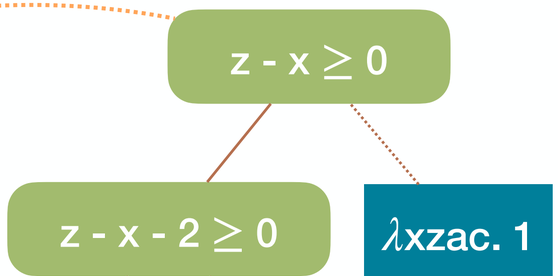
Loops

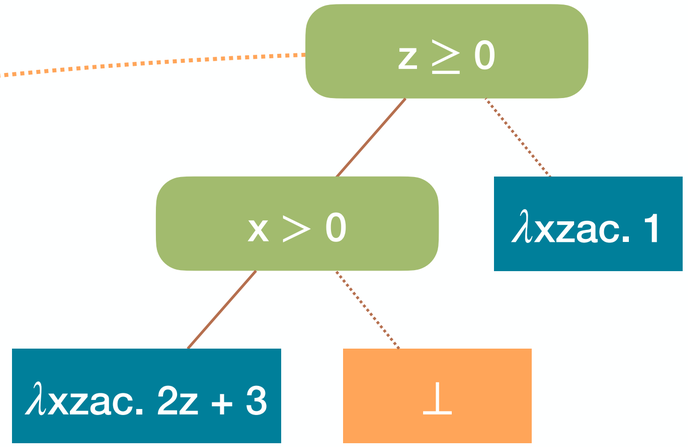
```
function f(x) {

  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```



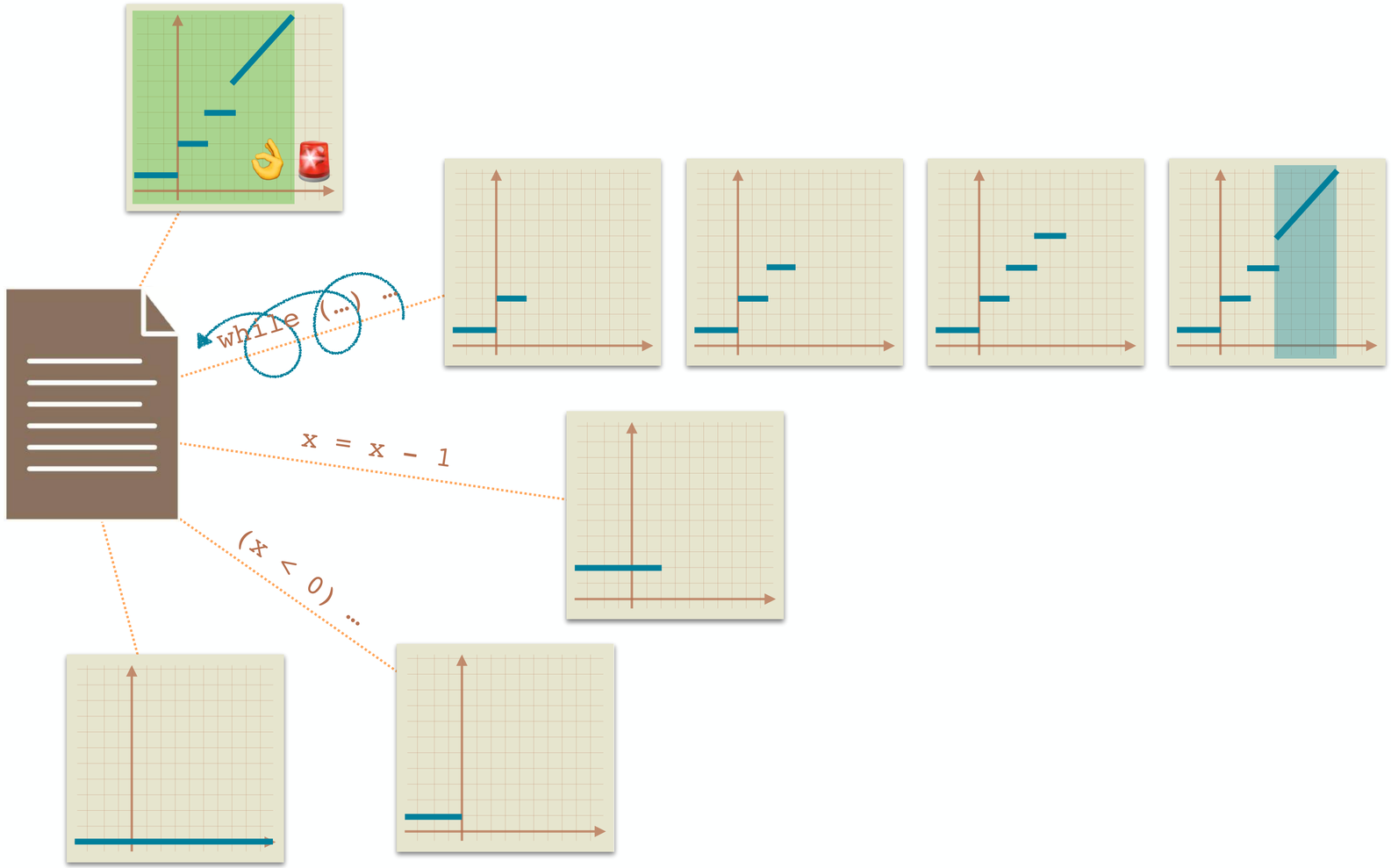








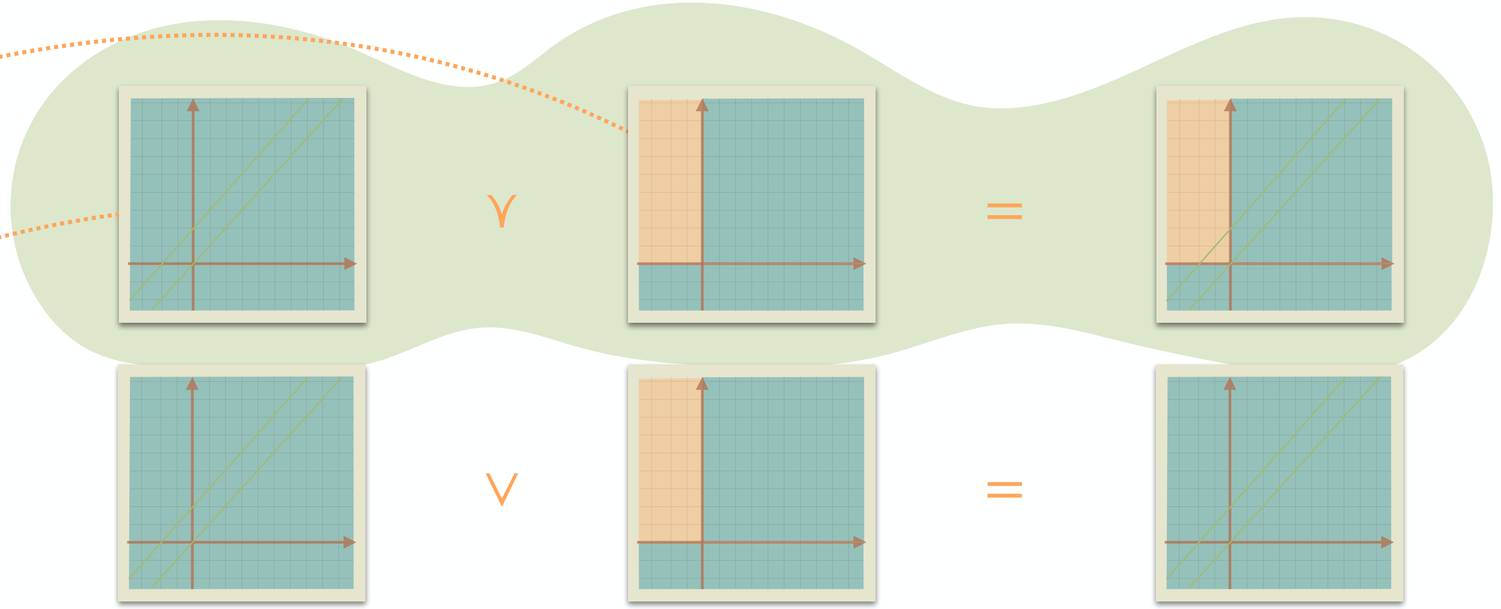
Termination Resilience Static Analysis



Termination Resilience Static Analysis

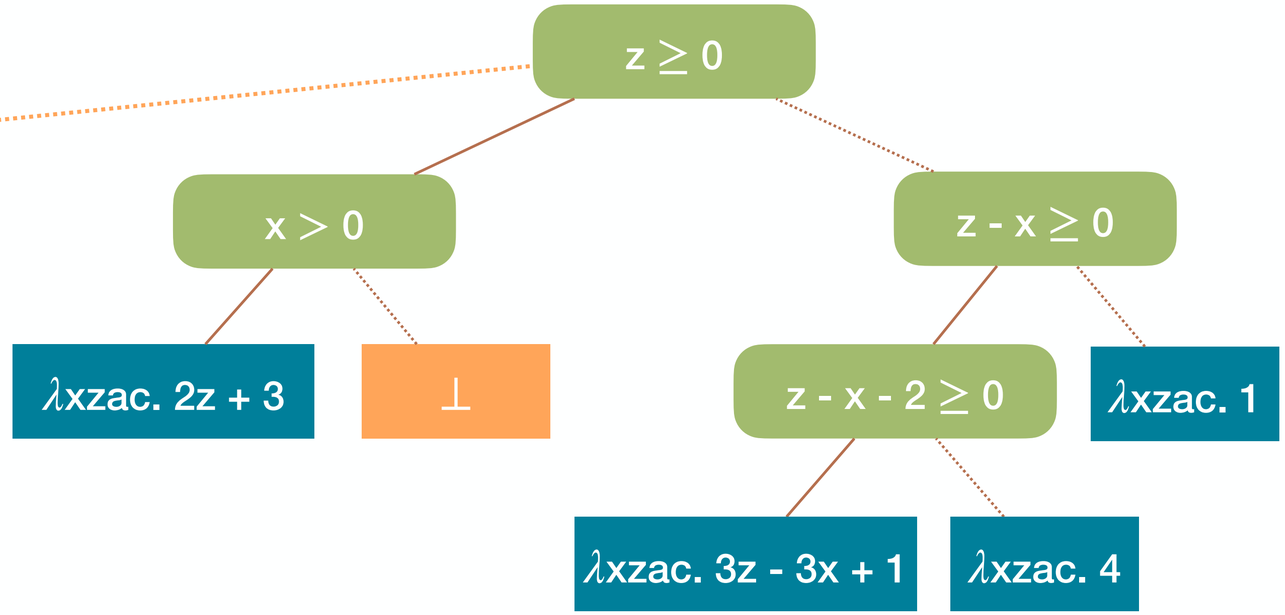
Approximation Join or Resilience Join?

```
function f(x) {
  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi
}11
```



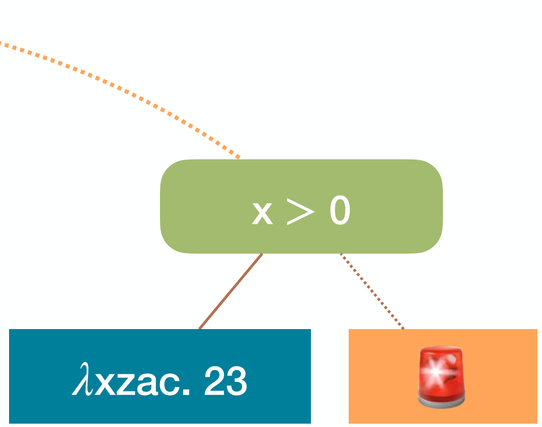
Termination Resilience Static Analysis

```
function f(x) {

  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi

}11
```



Termination Resilience Static Analysis

```
function f(x) {
  [1]a ← [-∞, +∞]
  [2]z ← 10
  [3]if (a*a ≥ 0) then
      while [4](z ≥ 0) do
        [5]z ← z - x
      od[6]
  else
      while [7](z ≥ x) do
        [8]c ←[-2, 1]
        [9]z ←z + c
      od[10]
  fi

}[11]
```



Termination Resilience Static Analysis

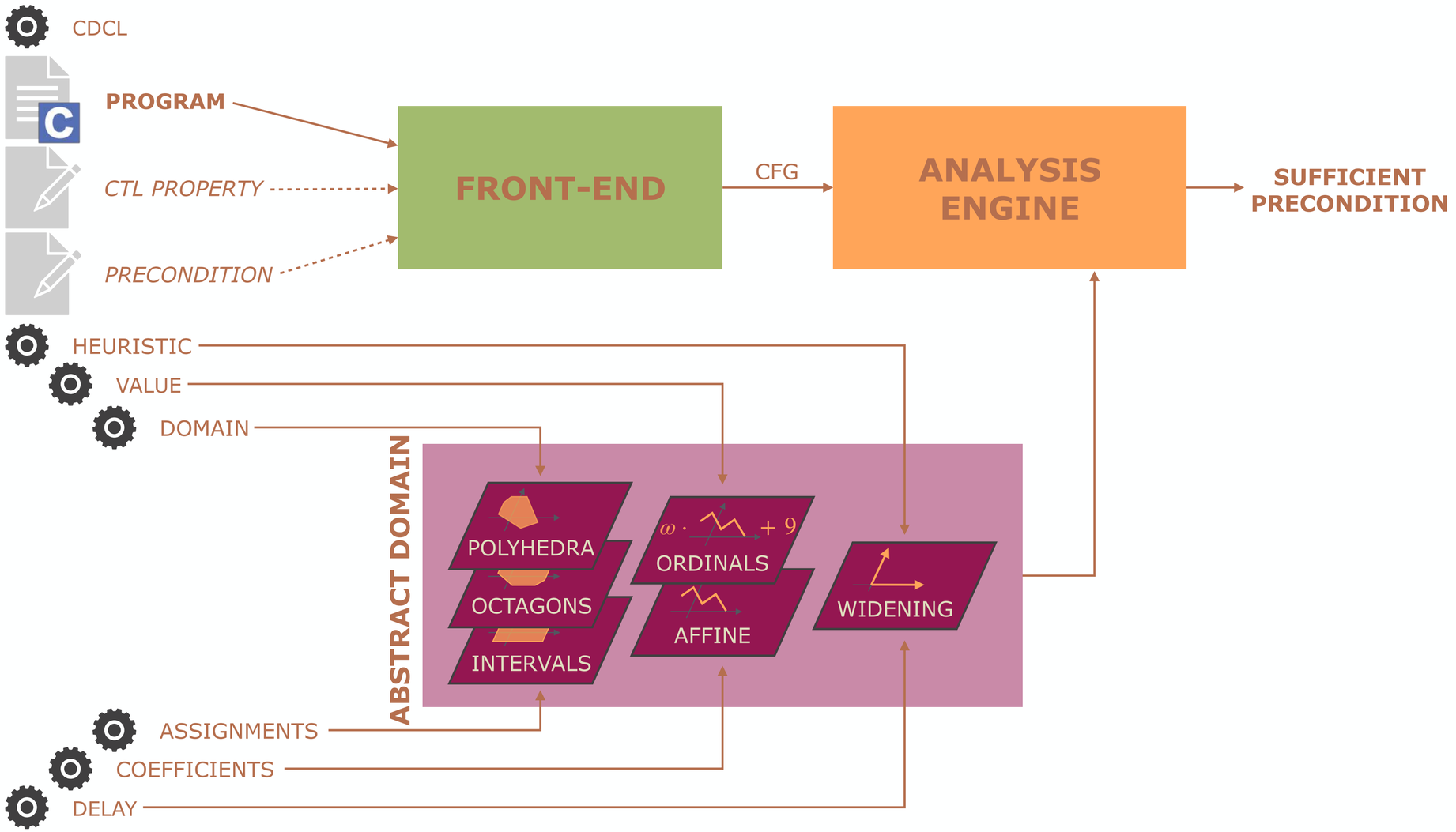
3-Step Recipe

practical tools
targeting specific programs

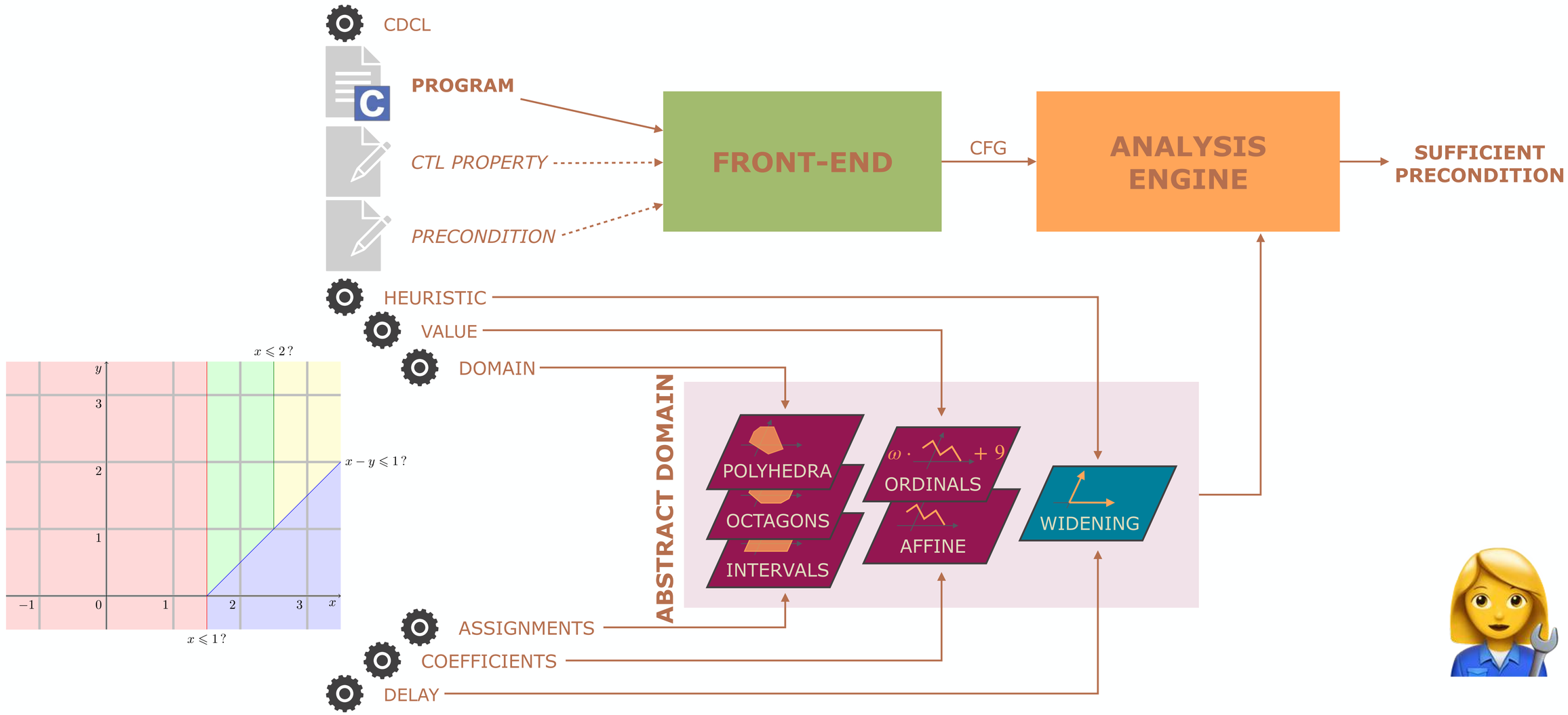
abstract semantics, abstract domains
algorithmic approaches to decide program properties

concrete semantics
mathematical models of the program behavior

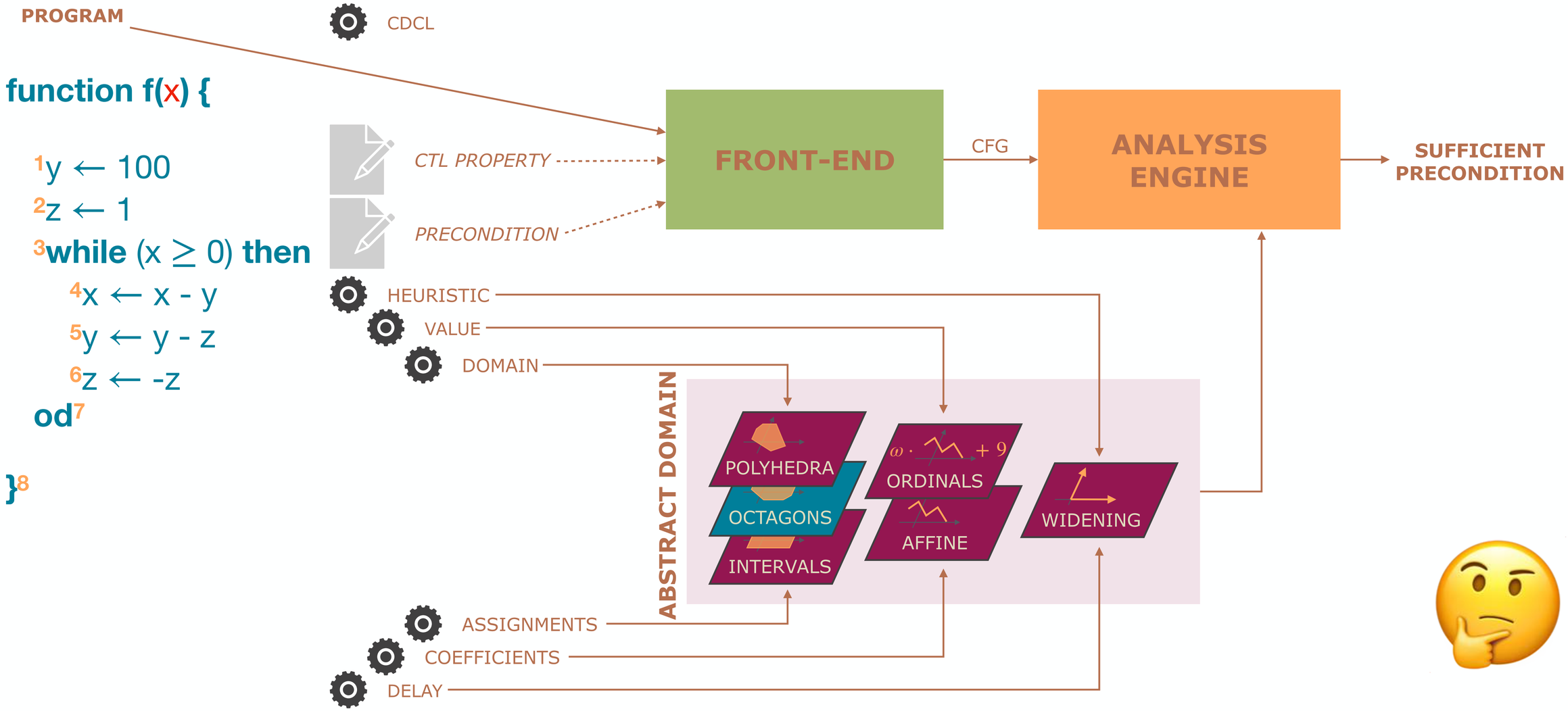
FuncTion


CDCL
PROGRAM
CTL PROPERTY
PRECONDITION
FRONT-END
CFG
ANALYSIS ENGINE
SUFFICIENT PRECONDITION
HEURISTIC
VALUE
DOMAIN
ABSTRACT DOMAIN
POLYHEDRA
OCTAGONS
INTERVALS
ORDINALS
AFFINE
WIDENING
ASSIGNMENTS
COEFFICIENTS
DELAY

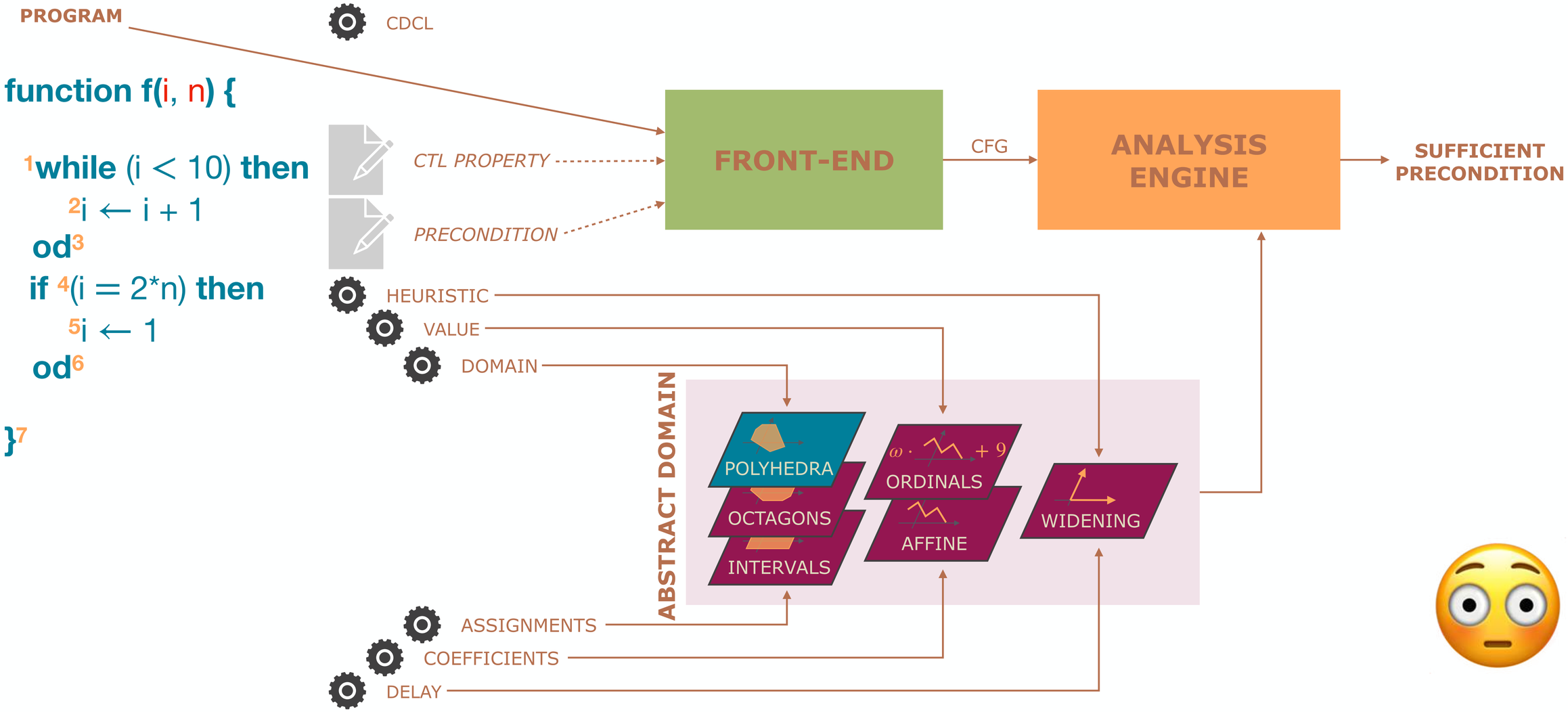
A Silent Bug in the Widening Operator



Non-Terminating (Forward) Analysis

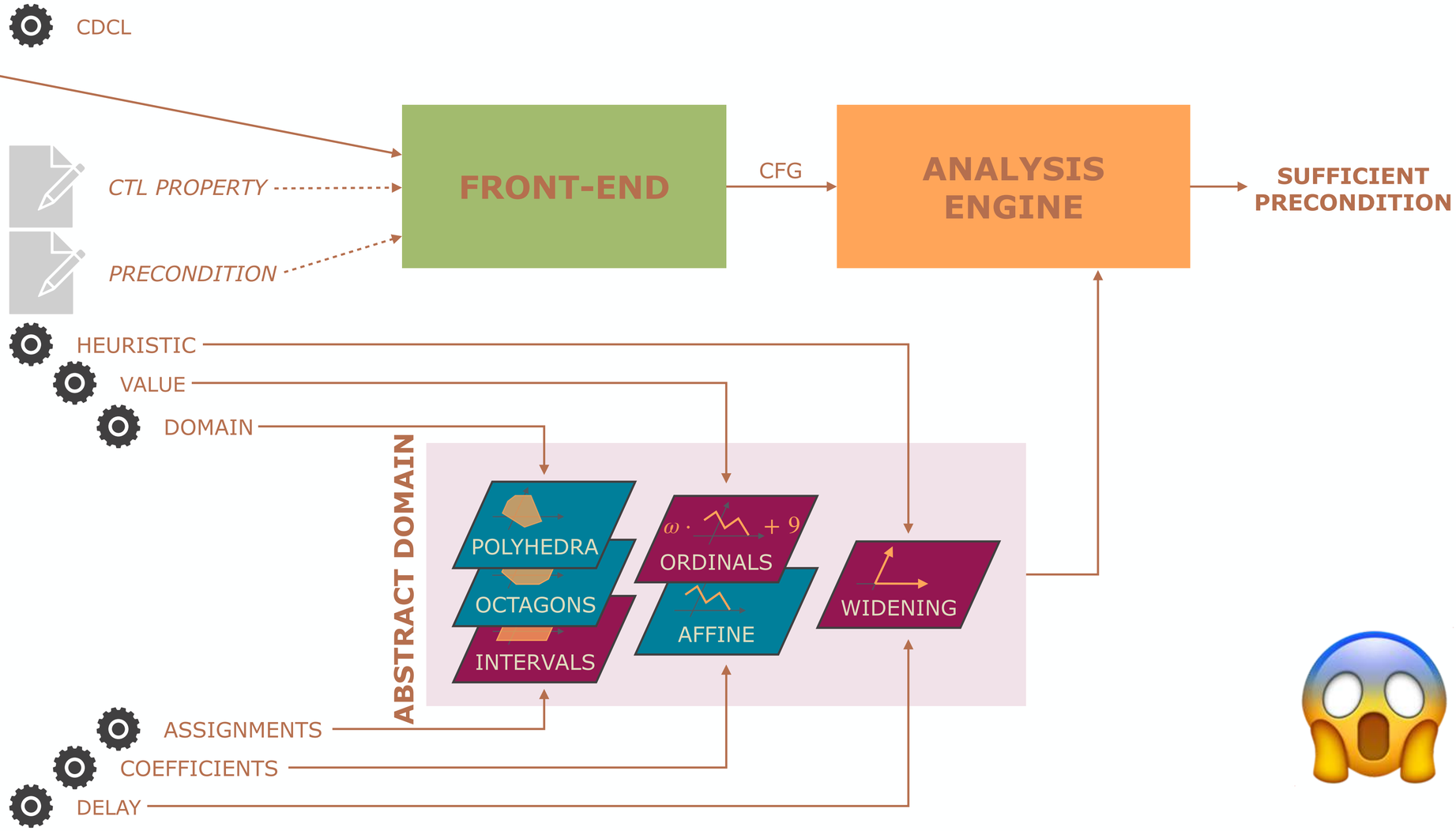


Unexpected Analysis Imprecision #1

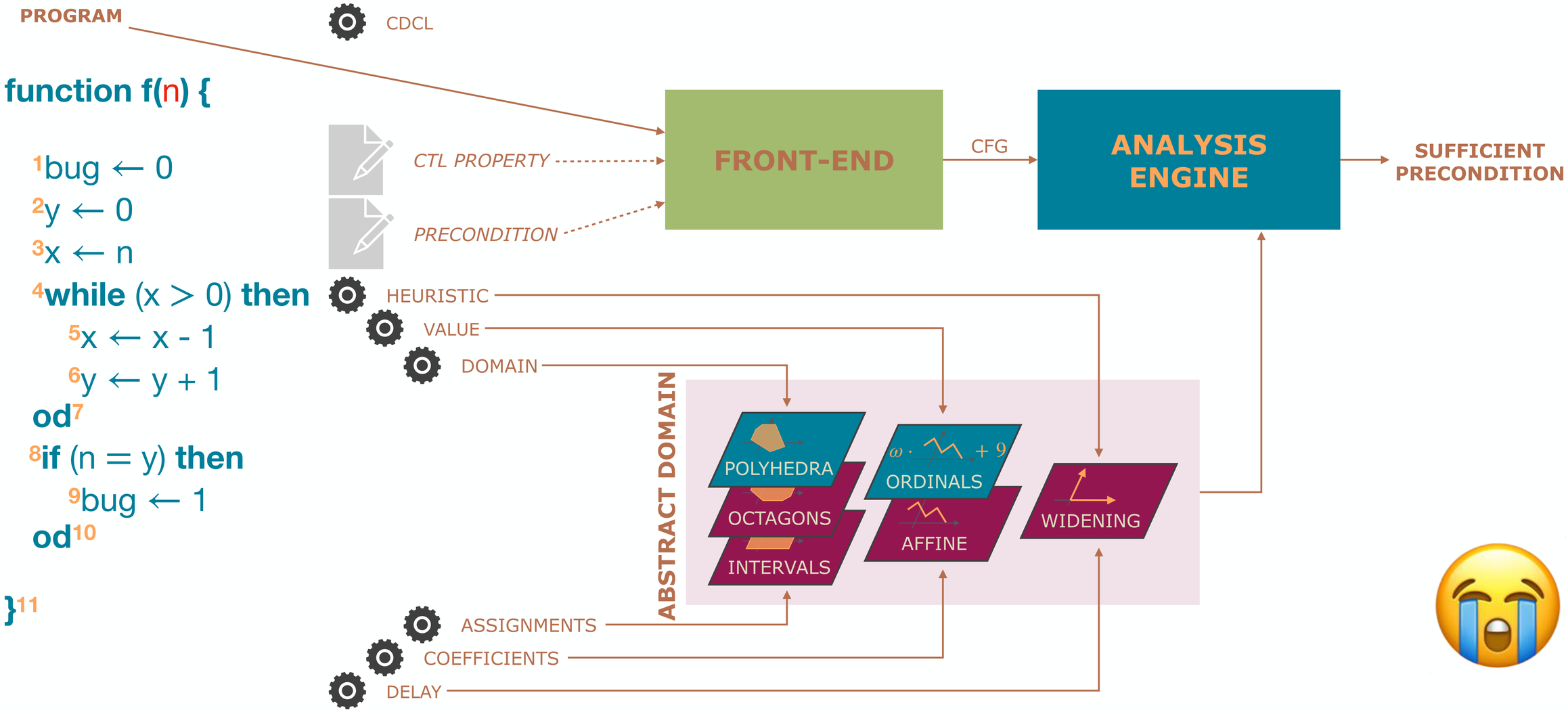


Unexpected Analysis Imprecision #2

```
function f(c) {
  [1]r ← 0
  [2]s ← 4
  [1]while (s > 0) then
      if [4](c ≥ s) then
          [5]c ← c - 1
          [5]r ← r + 1
          [2]s ← s - 1
      else
          [2]s ← s - 1
      endif
   od[3]
}[7]
```



Unexpected Analysis Imprecision #3




HELP

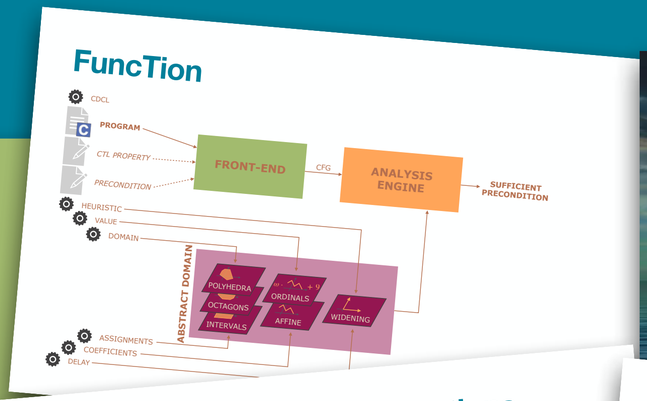
Termination Resilience Static Analysis

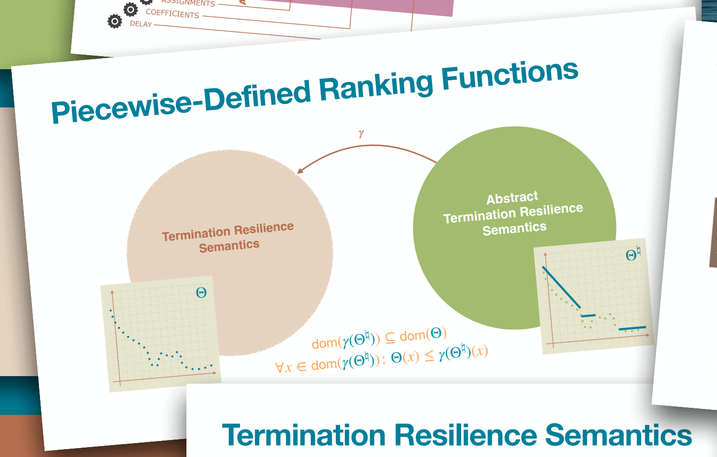
3-Step Recipe

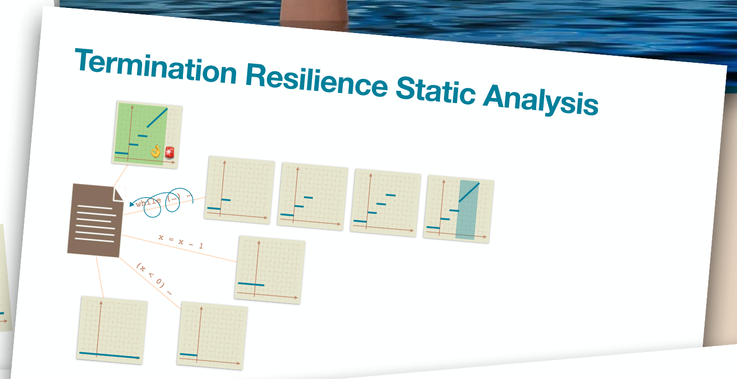
practical tools

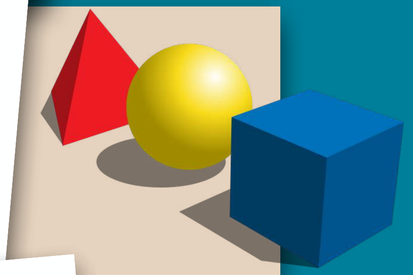
abstract semantics
abstract domains

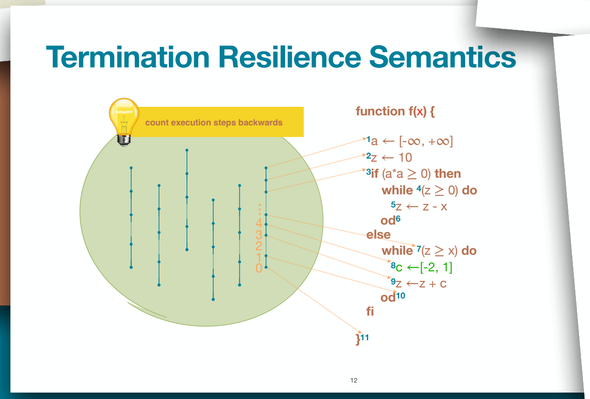
concrete semantics

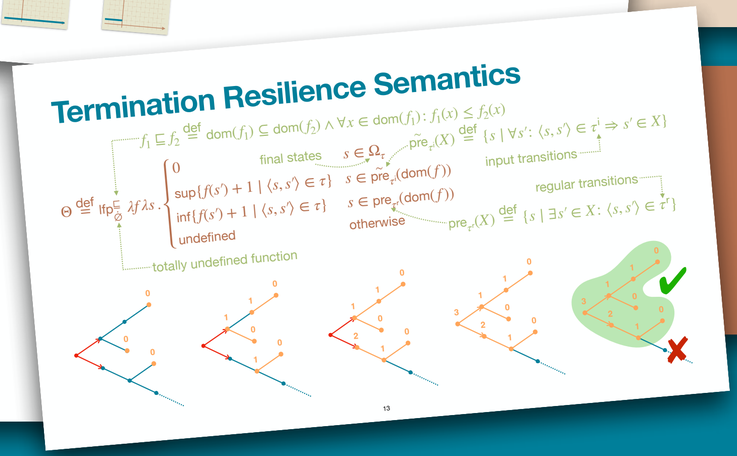












THANKS!