Termination Resilience Static Analysis

Seminari di Informatica, Università di Parma

Caterina Urban
Inria & École Normale Supérieure | Université PSL

Which Non-Termination Alarm is Worse?

```
function f(x) {

  [1]...
  [2]z ← 10
  [3]if ( ... ) then
      while [4](z ≥ 0) do          ← diverges when x = 0
        [5]z ← z - x
      od[6]
  else
      while [7](z ≥ x) do          ← diverges when c ≥ 0
        [8]c ←[-2, 1]              ← non-deterministic value choice
        [9]z ←z + c
      od[10]
  fi

}[11]
```

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 $x = 0$ (loop at 4)

← diverges when $c \geq 0$ (loop at 7)

← non-deterministic value choice (line 8)

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** (annotation on line 7: while (z ≥ x) do)

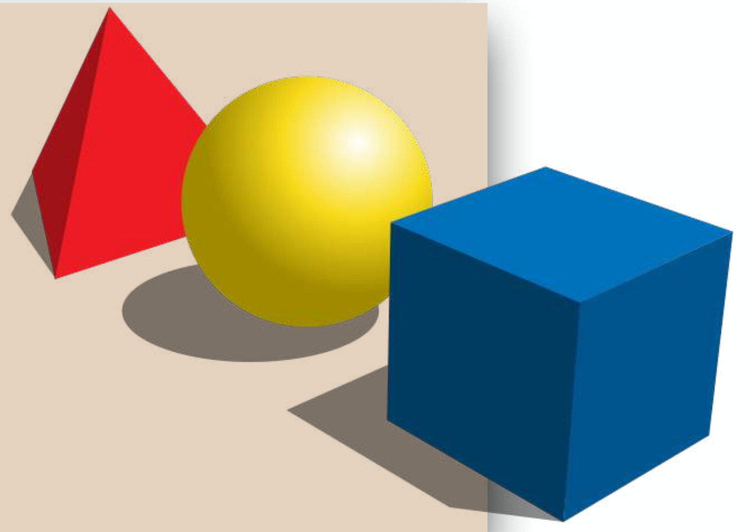
← angelic non-determinism (annotation on line 8: c ←[-2, 1])

Termination Resilience Static Analysis

3-Step Recipe

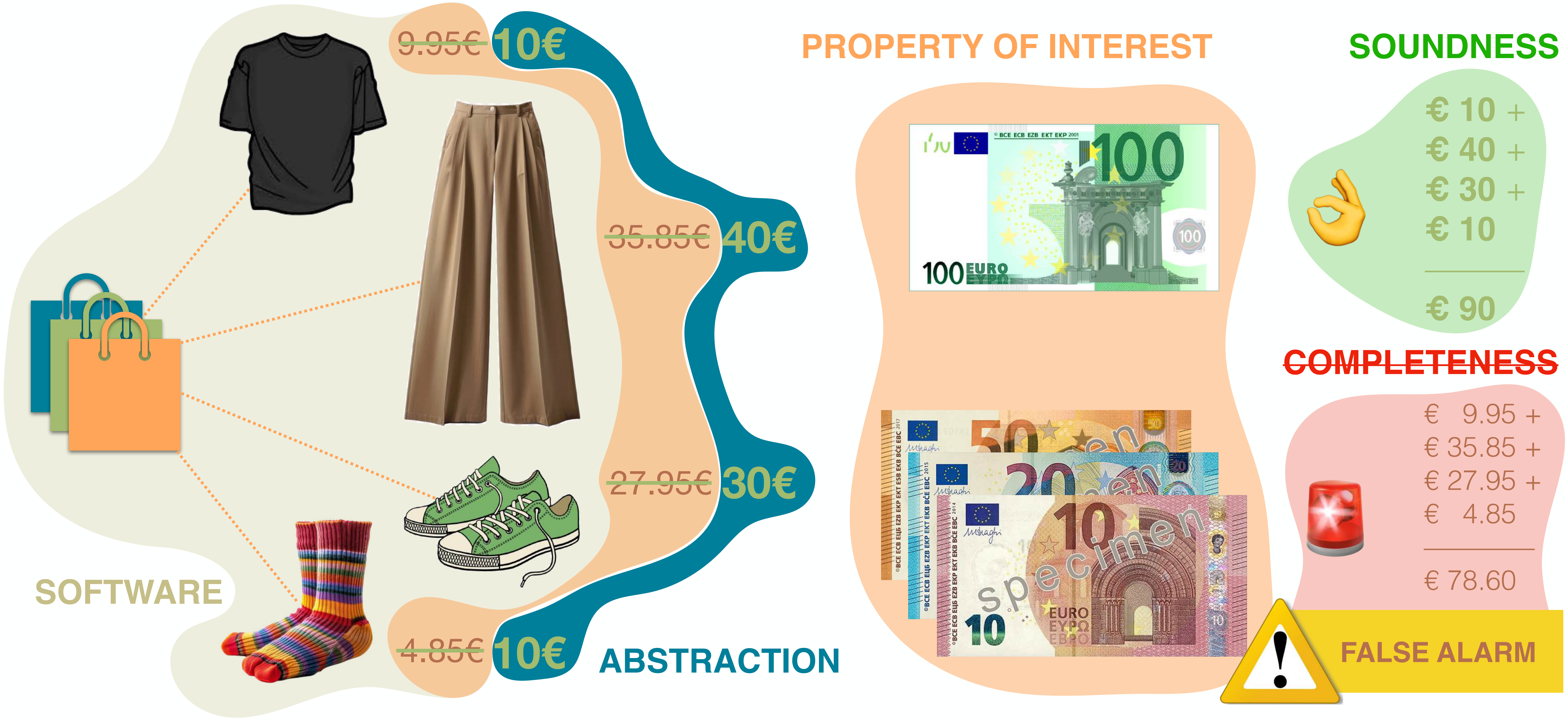
practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties



concrete semantics
mathematical models of the program behavior

Static Analysis by Abstract Interpretation



Termination Resilience Static Analysis

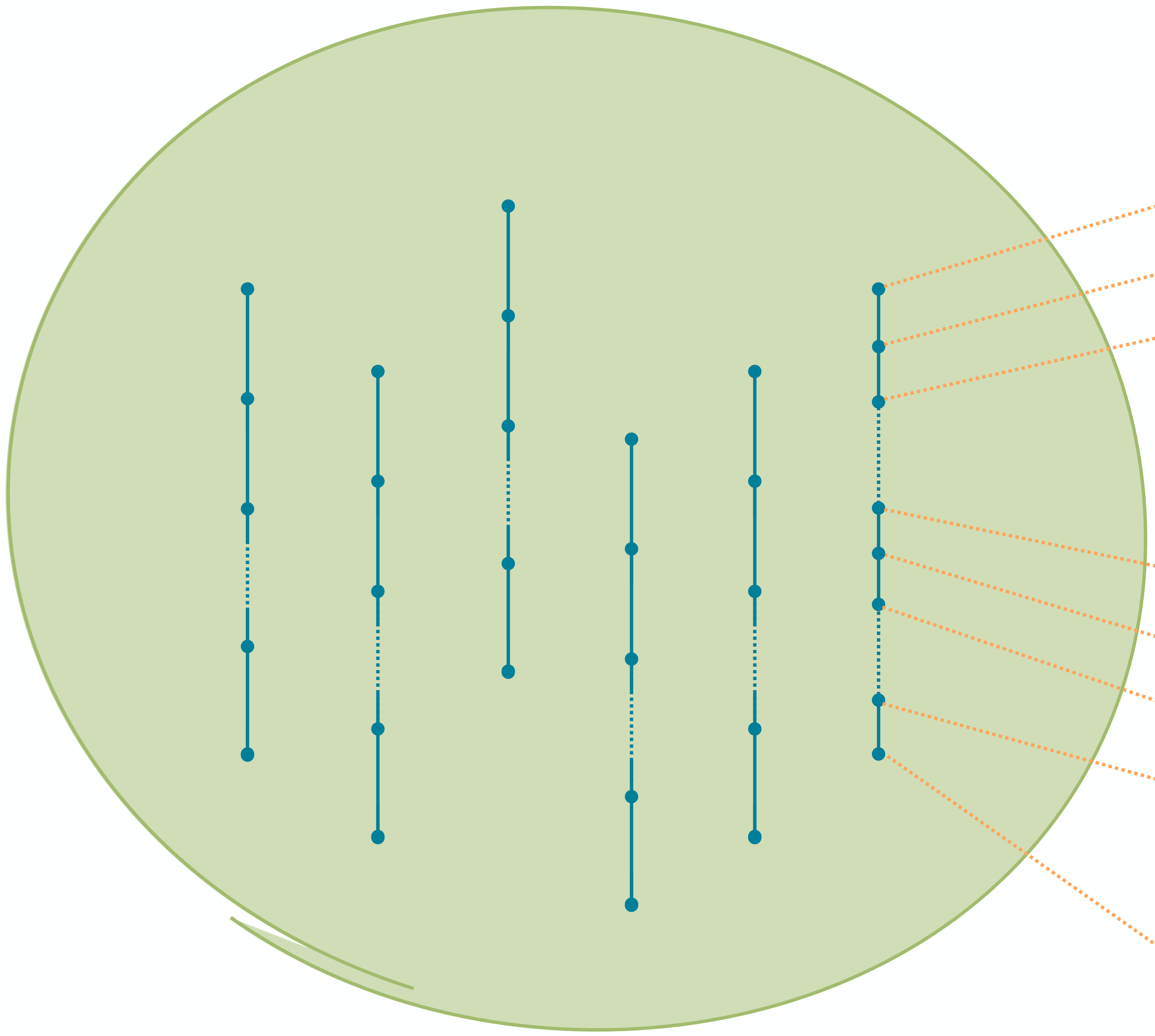
3-Step Recipe

practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties

concrete semantics
mathematical models of the program behavior

Trace Semantics

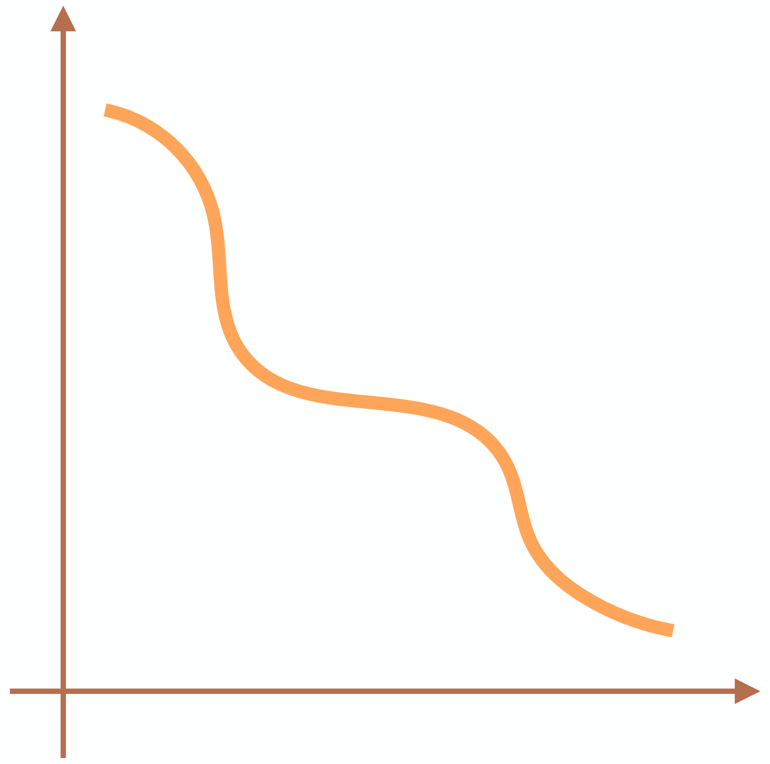



```
function f(x) {
  1a ← [-∞, +∞]
  2z ← 10
  3if (a*a ≥ 0) then
      while 4(z ≥ 0) do
        5z ← z - x
      od6
  else
      while 7(z ≥ x) do
        8c ←[-2, 1]
        9z ←z + c
      od10
  fi
}11
```

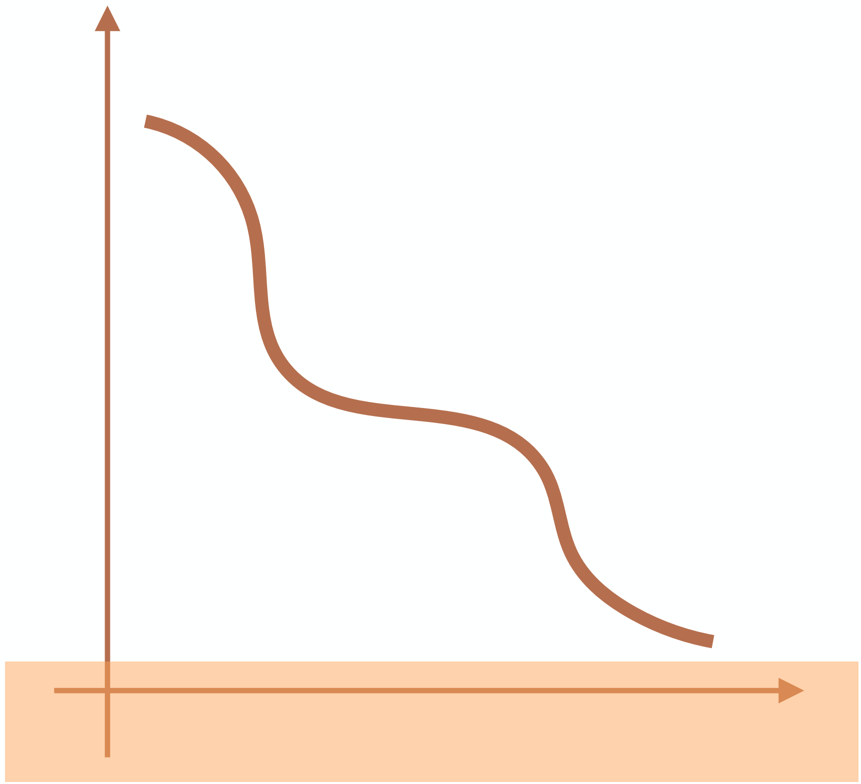
Ranking Functions

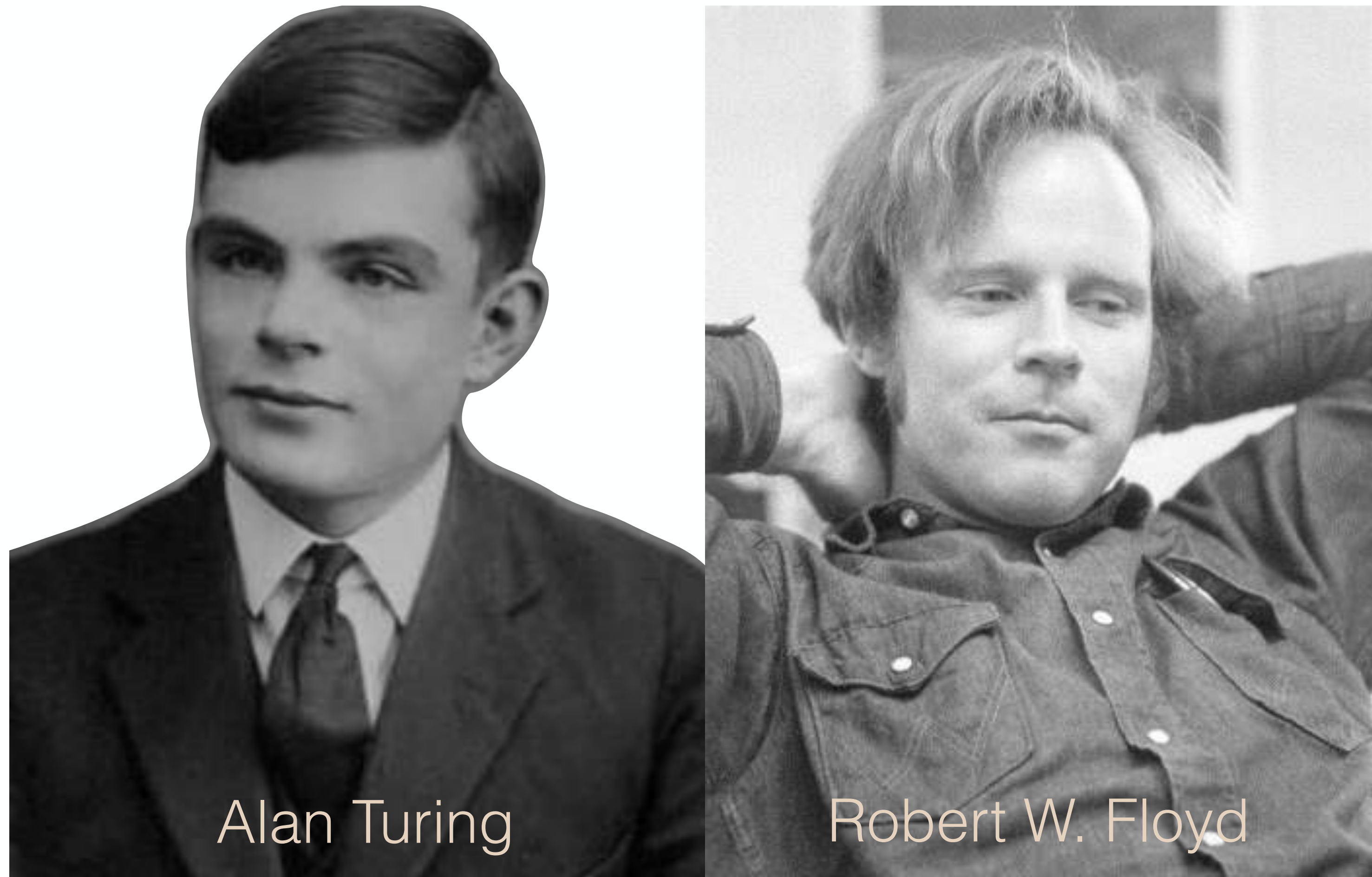
Traditional Method for Proving Termination

strictly decreasing along the execution of a program…

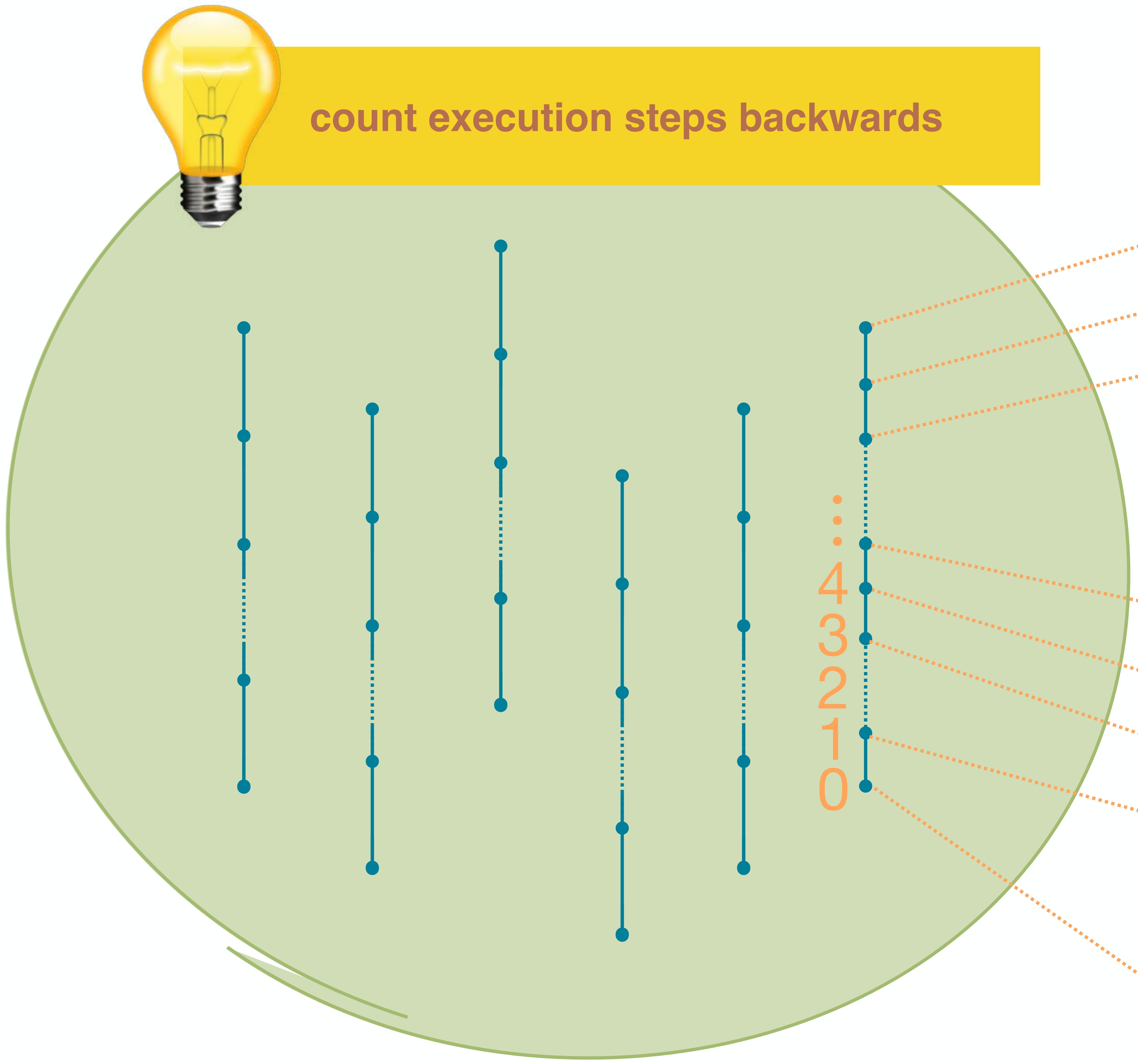


…and **well-founded**





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

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

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

final states: $s \in \Omega_\tau$

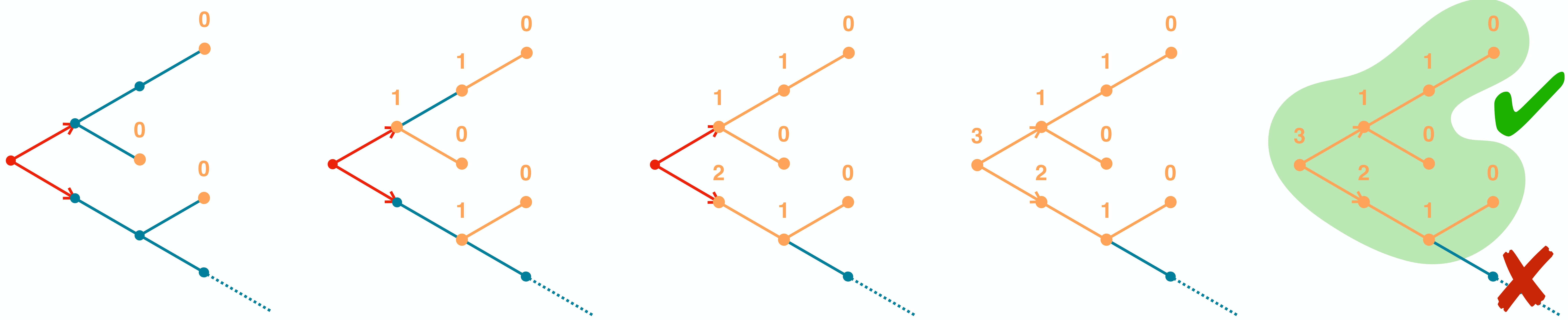
$$\widetilde{\text{pre}}_{\tau^{\text{i}}}(X) \overset{\text{def}}{=} \{s \mid \forall s' \colon \langle s, s' \rangle \in \tau^{\text{i}} \Rightarrow s' \in X\}$$

input transitions: τ^{i}

regular transitions: τ^{r}

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

totally undefined function: $\dot\emptyset$



Termination Resilience Semantics

$$\Theta \stackrel{\text{def}}{=} \text{lfp}^{\sqsubseteq}_{\emptyset}\ \lambda f\, \lambda s\,.\begin{cases} 0 & s \in \Omega_\tau \\ \text{sup}\{f(s') + 1 \mid \langle s, s' \rangle \in \tau\} & s \in \widetilde{\text{pre}}_{\tau^{\text{i}}}(\text{dom}(f)) \\ \text{sup}\{f(s') + 1 \mid \langle s, s' \rangle \in \tau\} & s \in \text{pre}_{\tau^{\text{r}}}(\text{dom}(f)) \\ \text{undefined} & \text{otherwise} \end{cases}$$

the existence of the fixpoint is not guaranteed

```
1x ← [-∞, +∞]
while 2(x ≠ 0) do
    3x ← [-∞, +∞]
od4
```

$$\lambda x\,.\begin{cases} 1 & x = 0 \\ \text{undefined} & \text{otherwise} \end{cases} \qquad \lambda x\,.\begin{cases} 3 & x = 0 \\ \text{undefined} & \text{otherwise} \end{cases} \qquad \dots$$

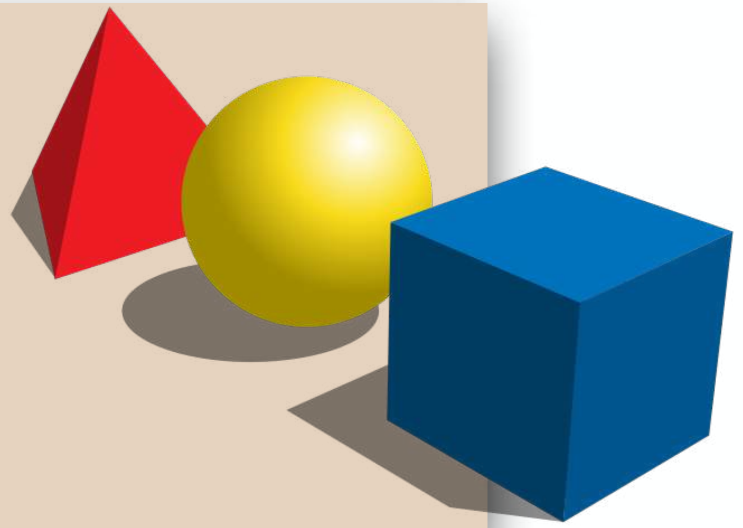
$\lambda x.0$

Termination Resilience Static Analysis

3-Step Recipe

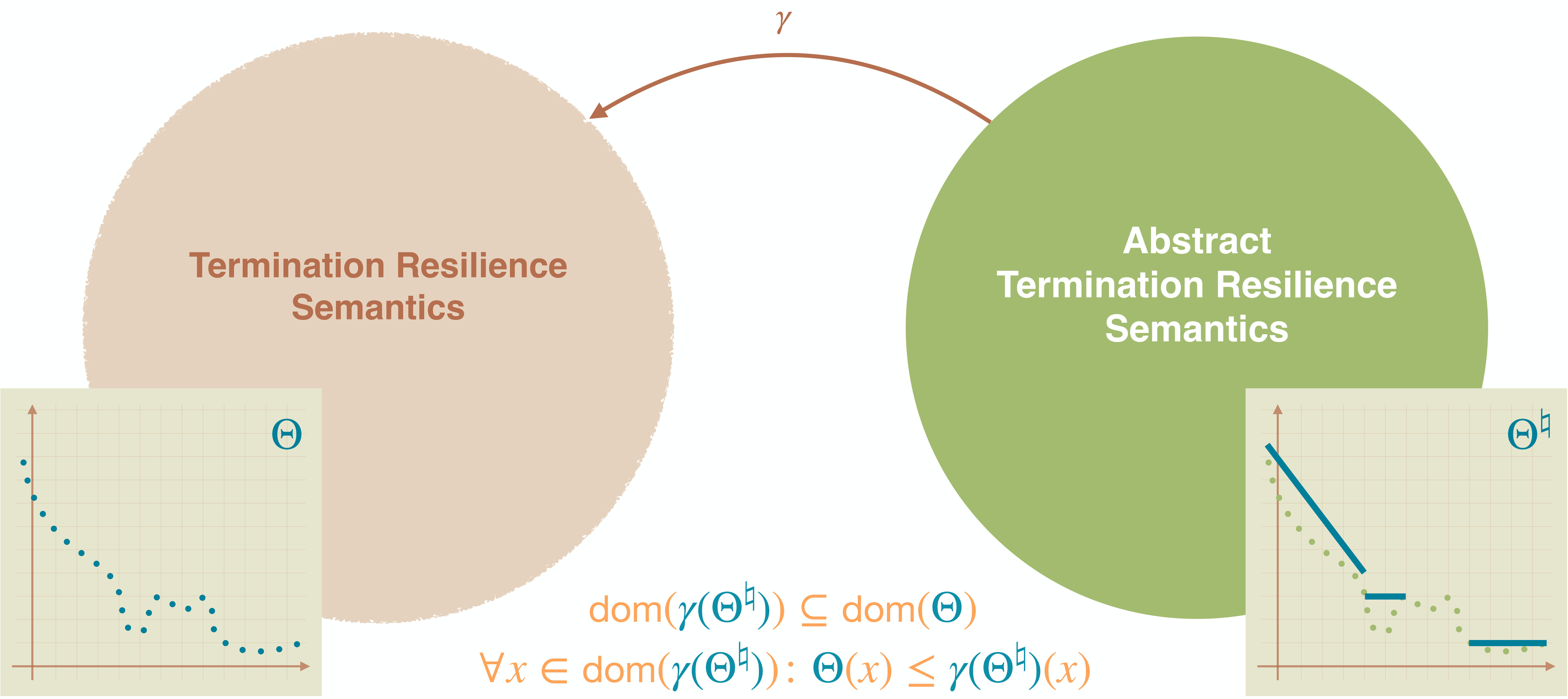
practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties



concrete semantics
mathematical models of the program behavior

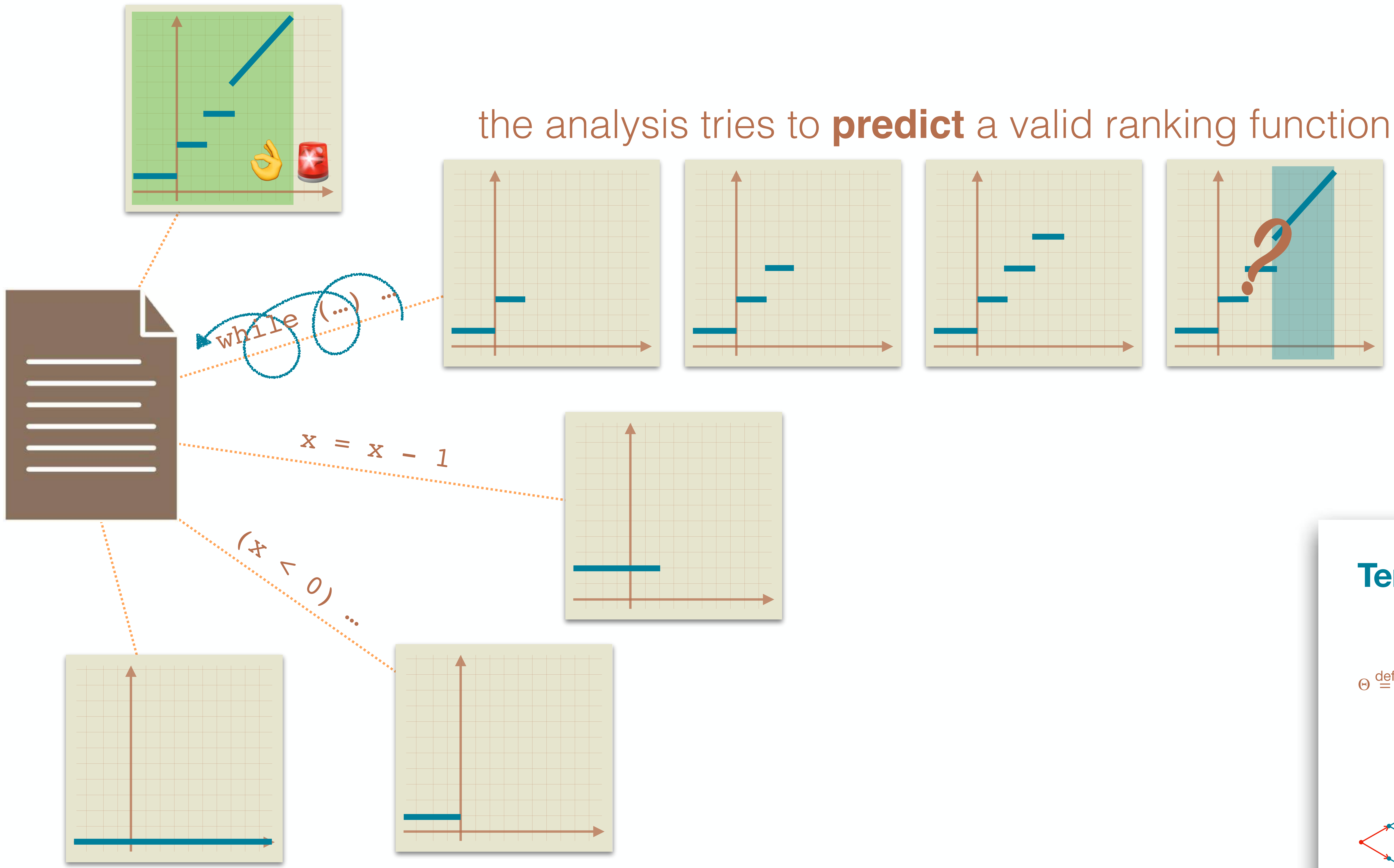
Piecewise-Defined Ranking Functions

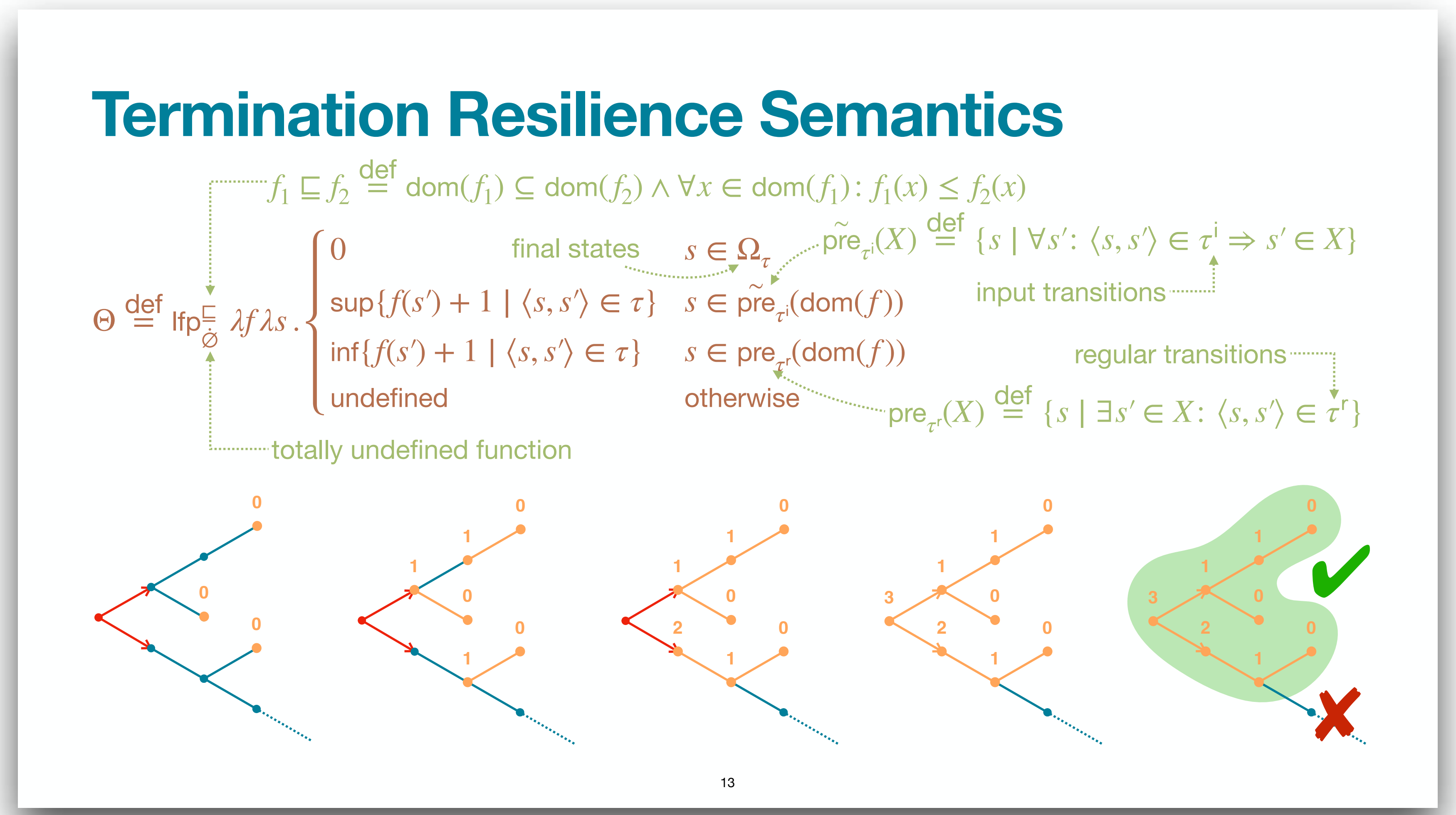


$$\text{dom}(\gamma(\Theta^\natural)) \subseteq \text{dom}(\Theta)$$

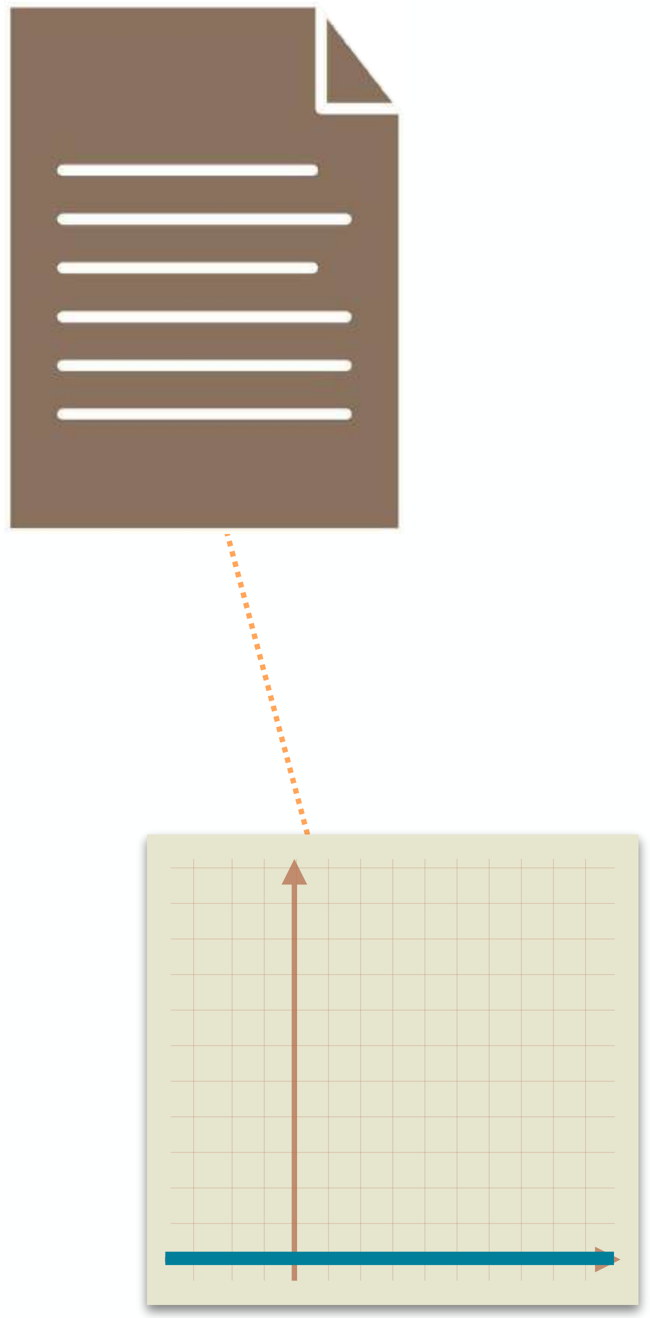
$$\forall x \in \text{dom}(\gamma(\Theta^\natural)) : \Theta(x) \leq \gamma(\Theta^\natural)(x)$$

Termination Resilience Static Analysis





Termination Resilience Static Analysis

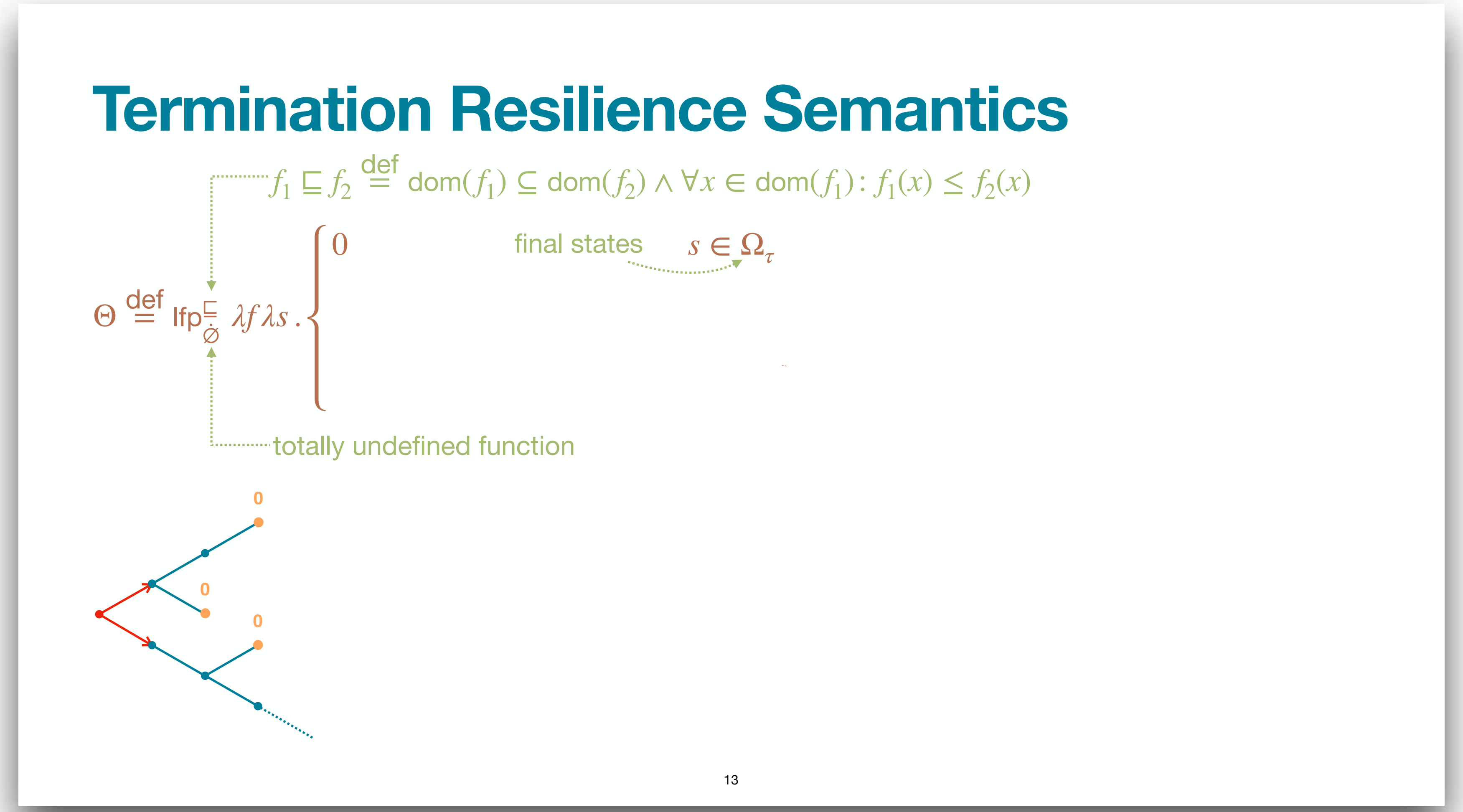


Termination Resilience Static Analysis

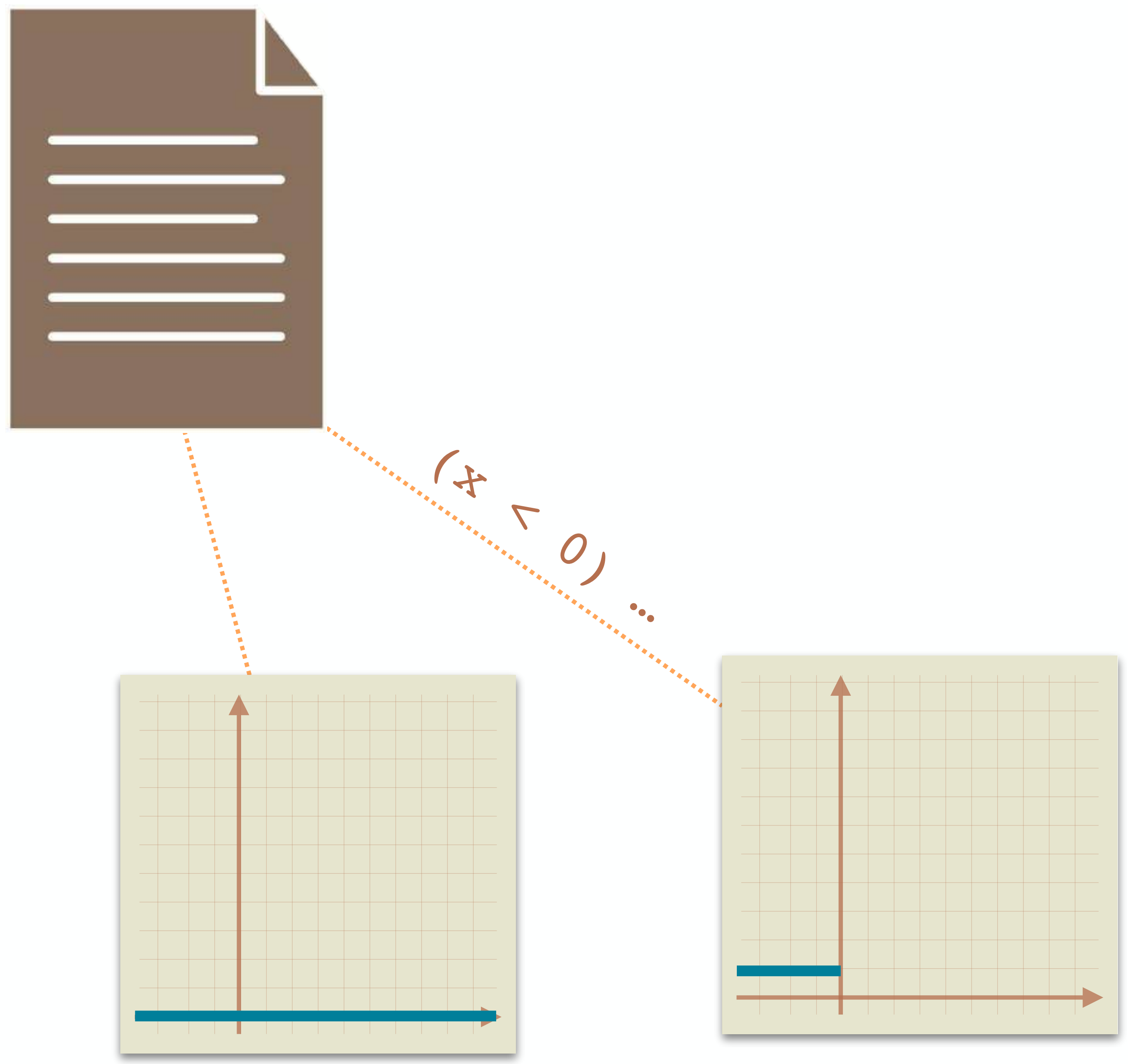
Static Backward Analysis

```
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
```

$\lambda xzac.\ 0$



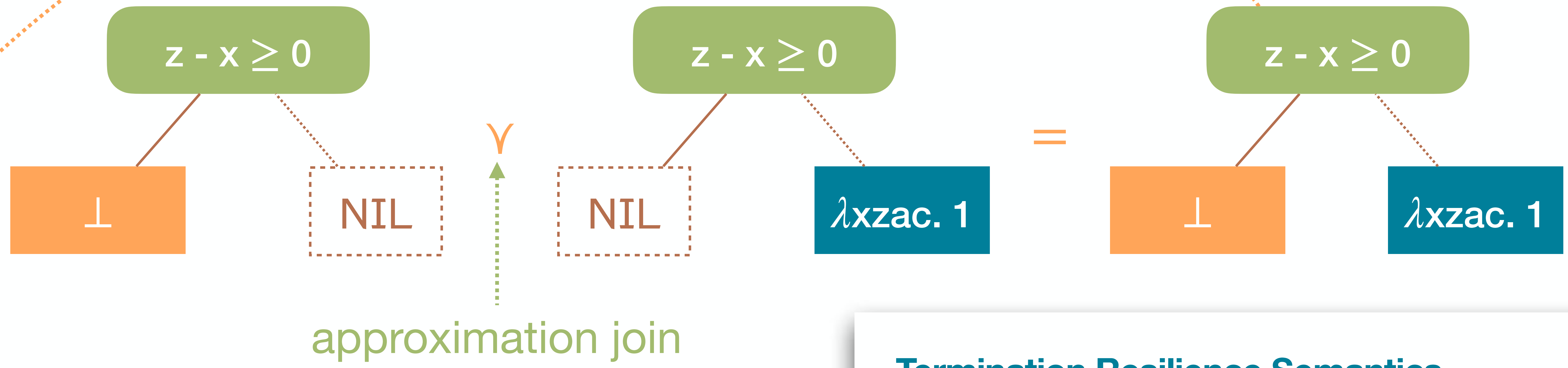
Termination Resilience Static Analysis

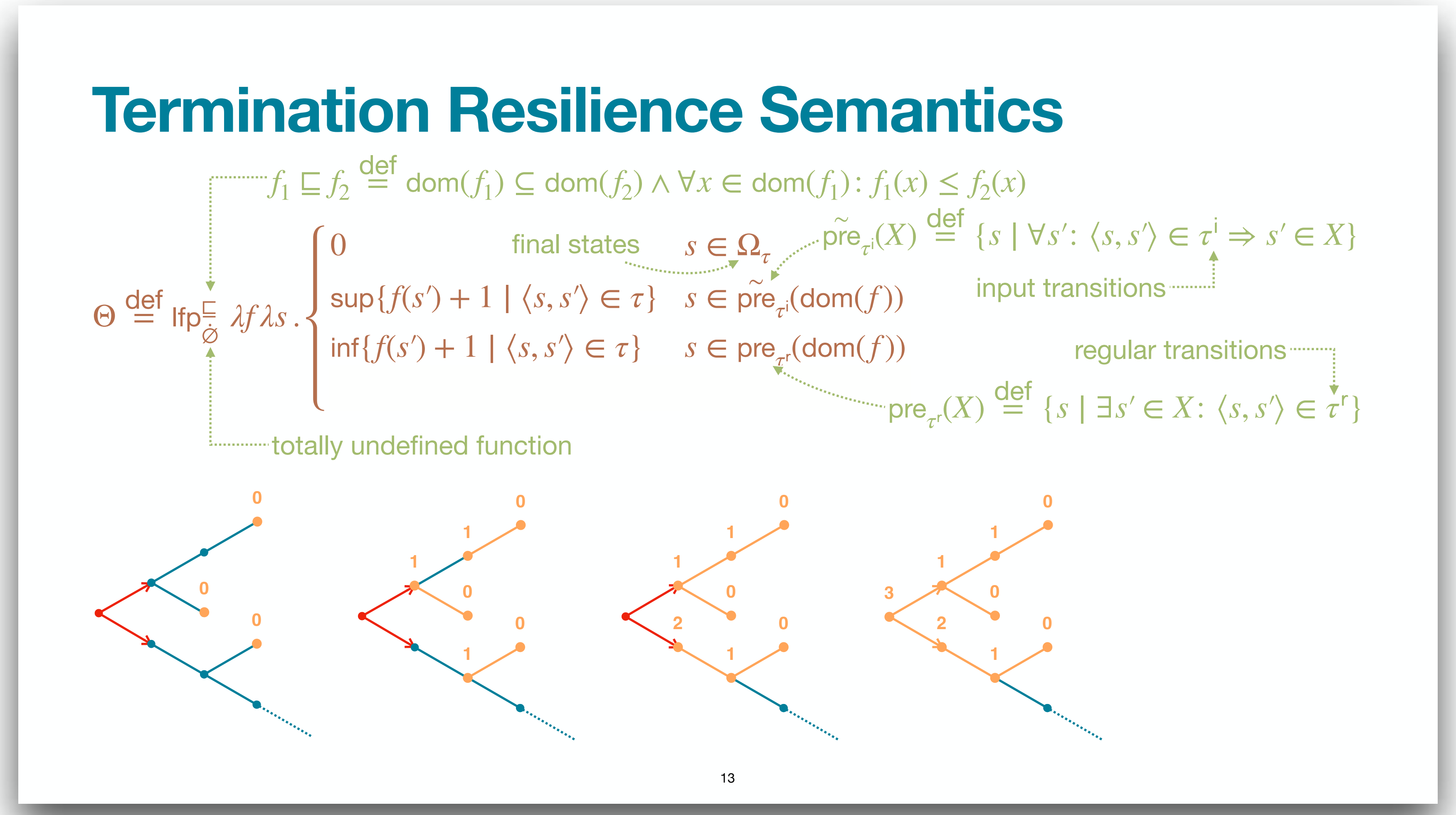


Termination Resilience Static Analysis

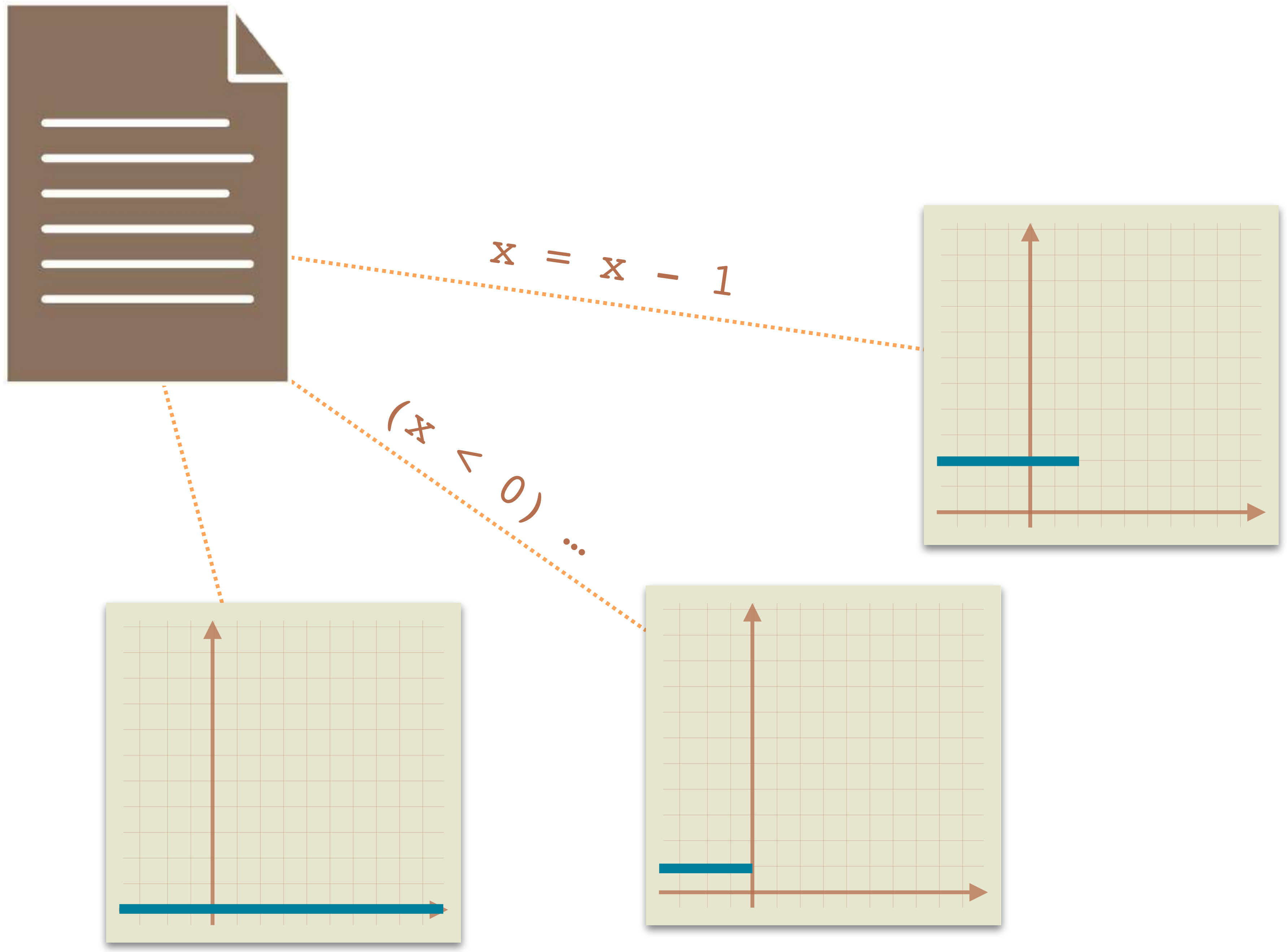
Boolean Conditions

```
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
```





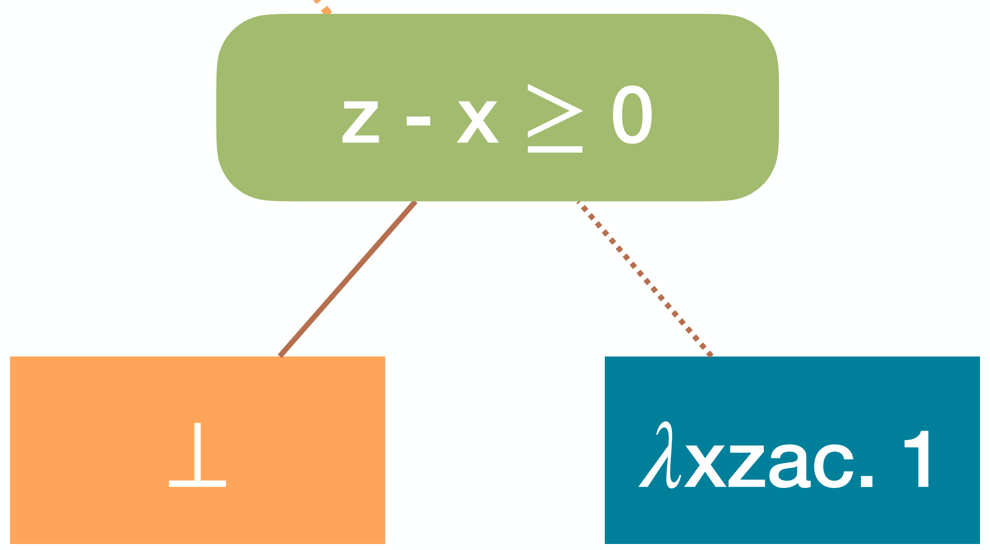
Termination Resilience Static Analysis

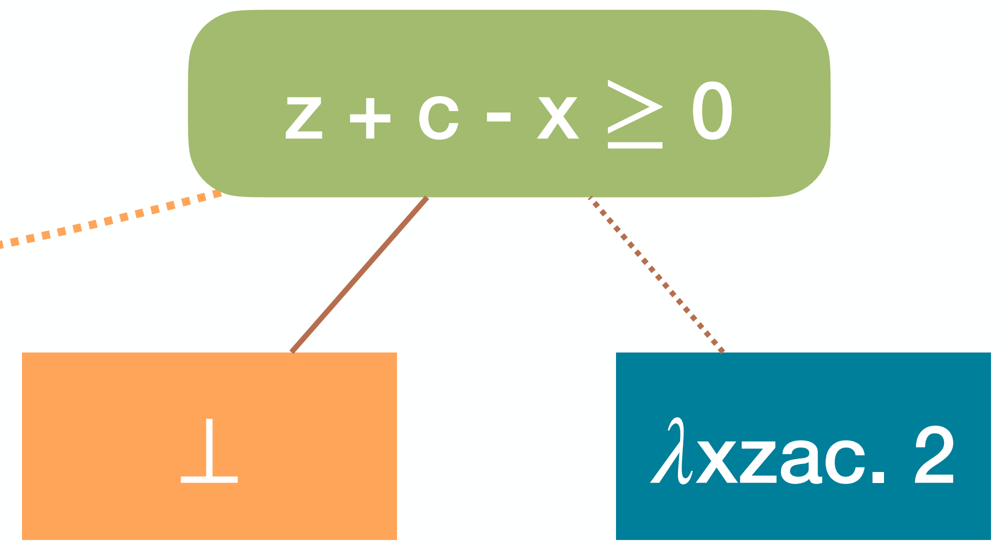


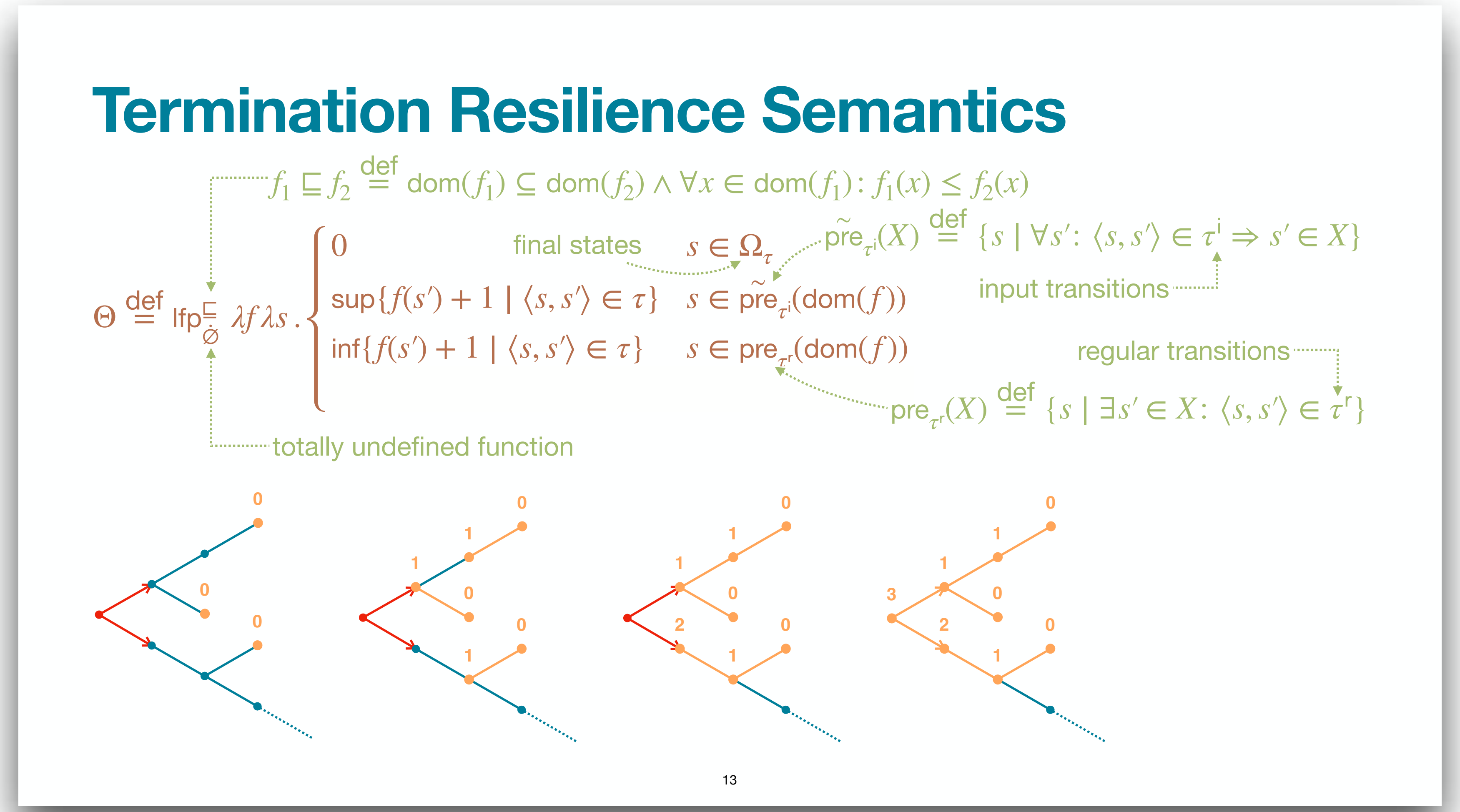
Termination Resilience Static Analysis

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
```



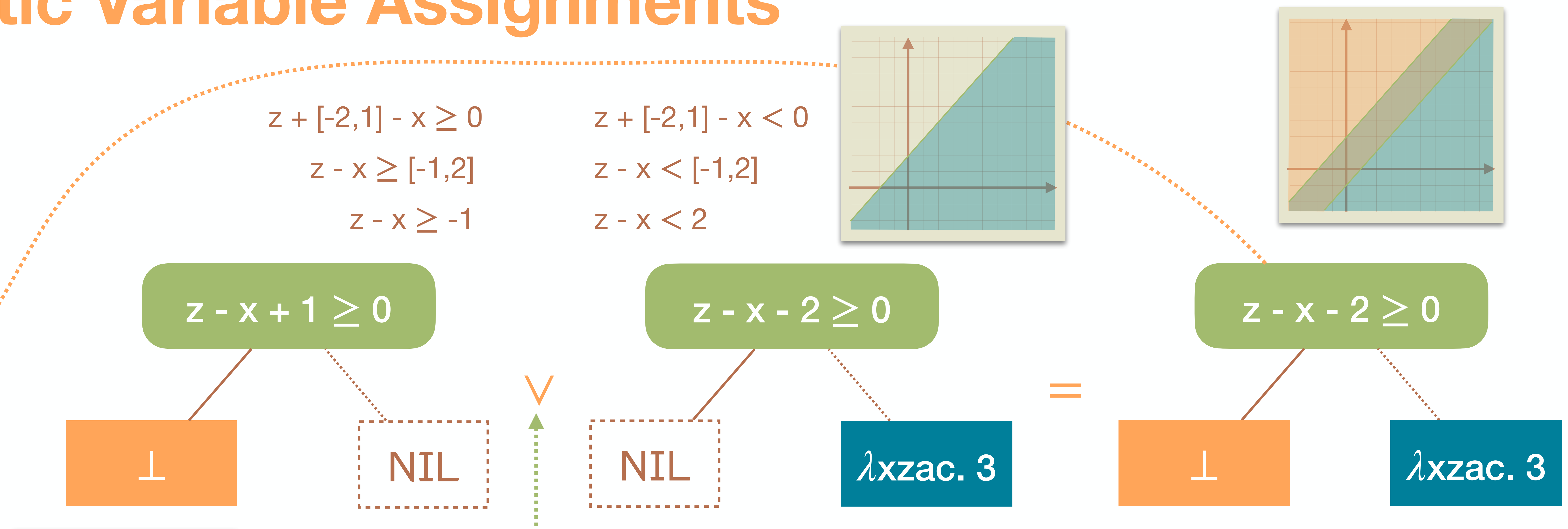


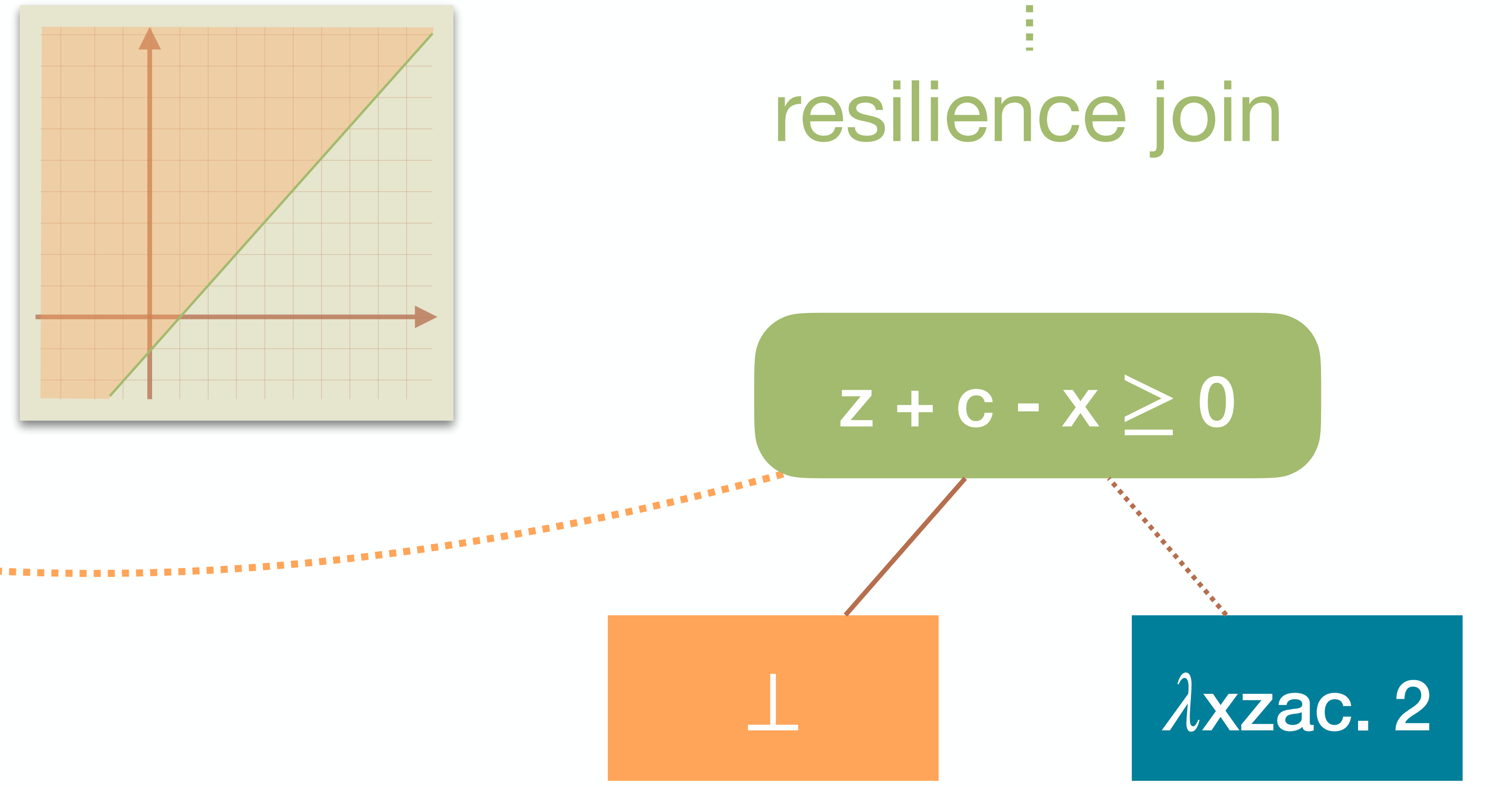


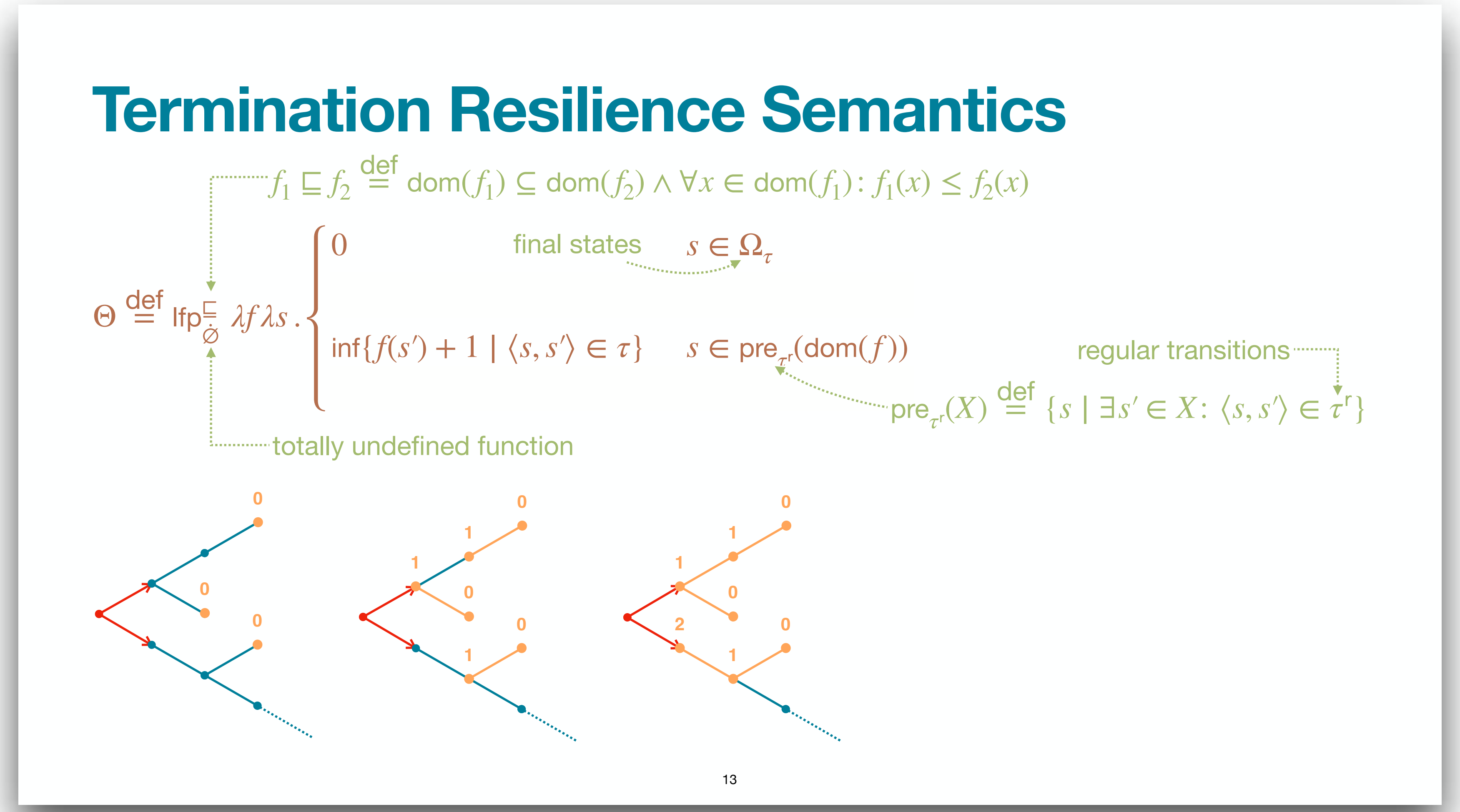
Termination Resilience Static Analysis

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
```

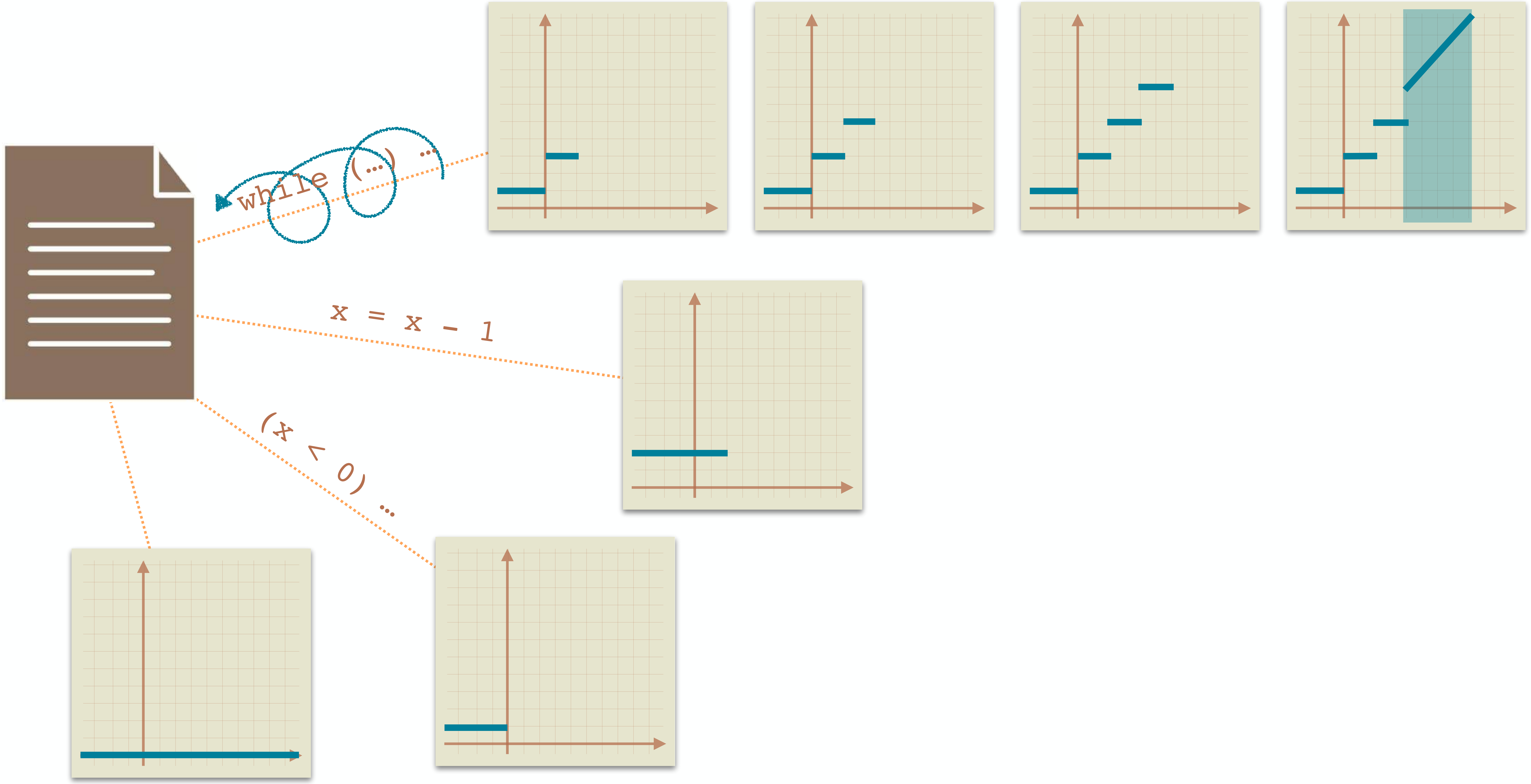






Termination Resilience Static Analysis

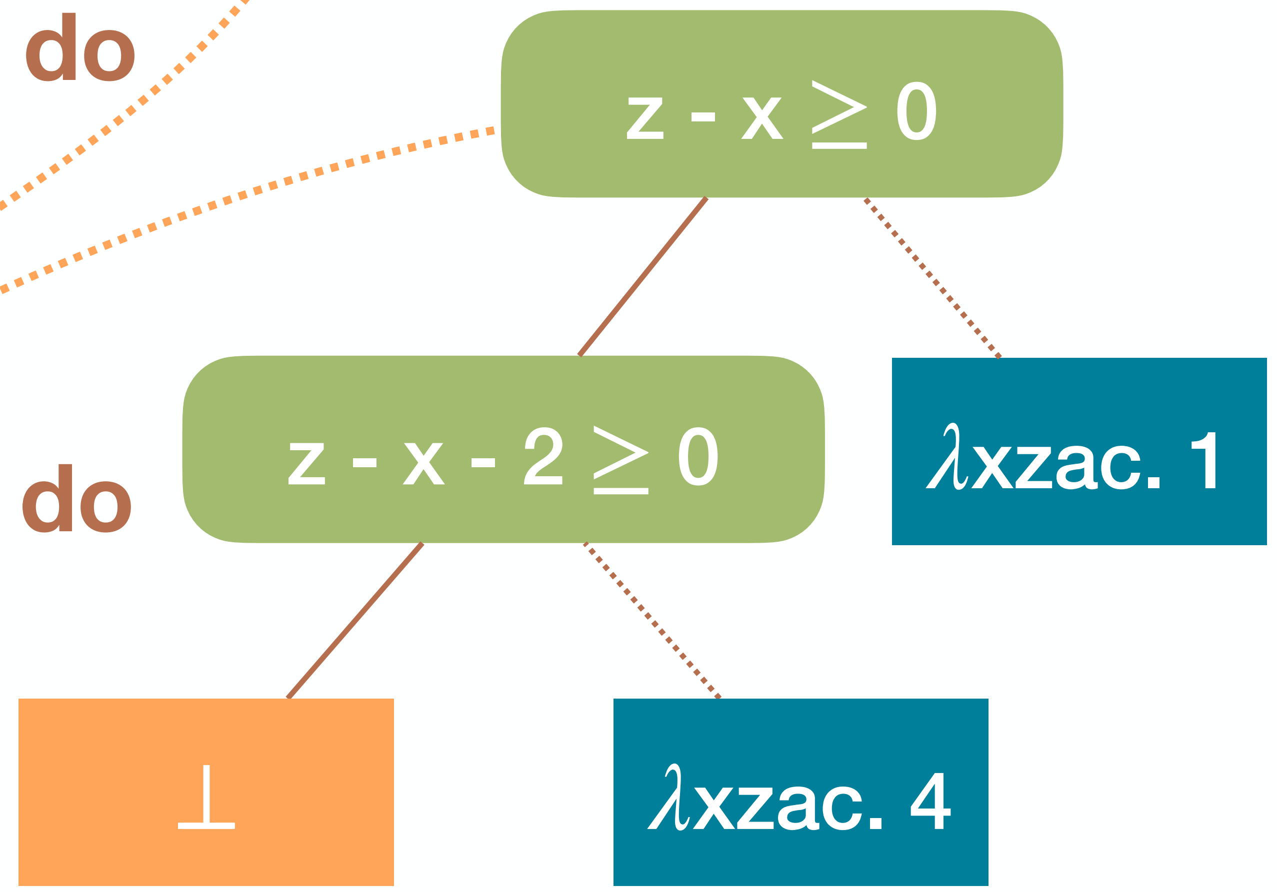
the analysis tries to **predict** a valid ranking function

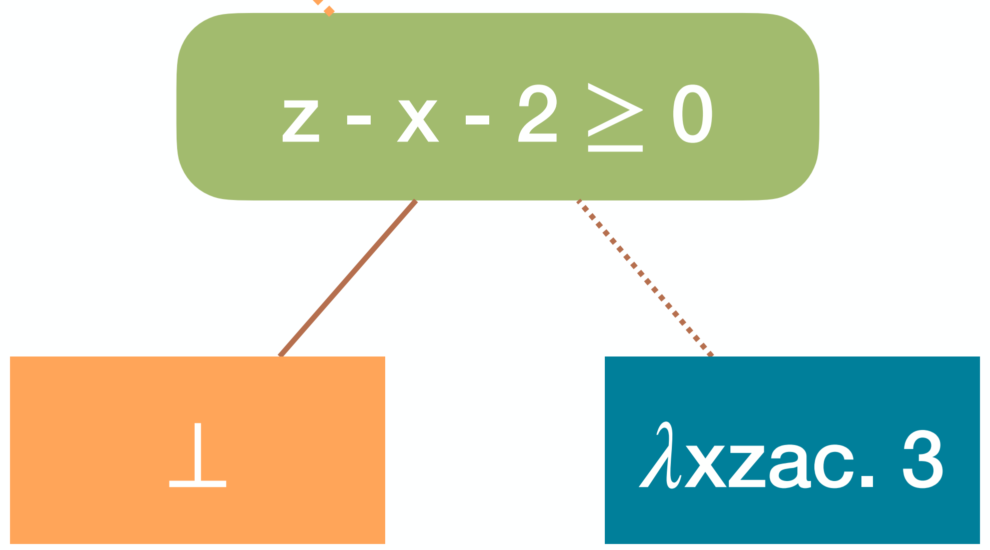


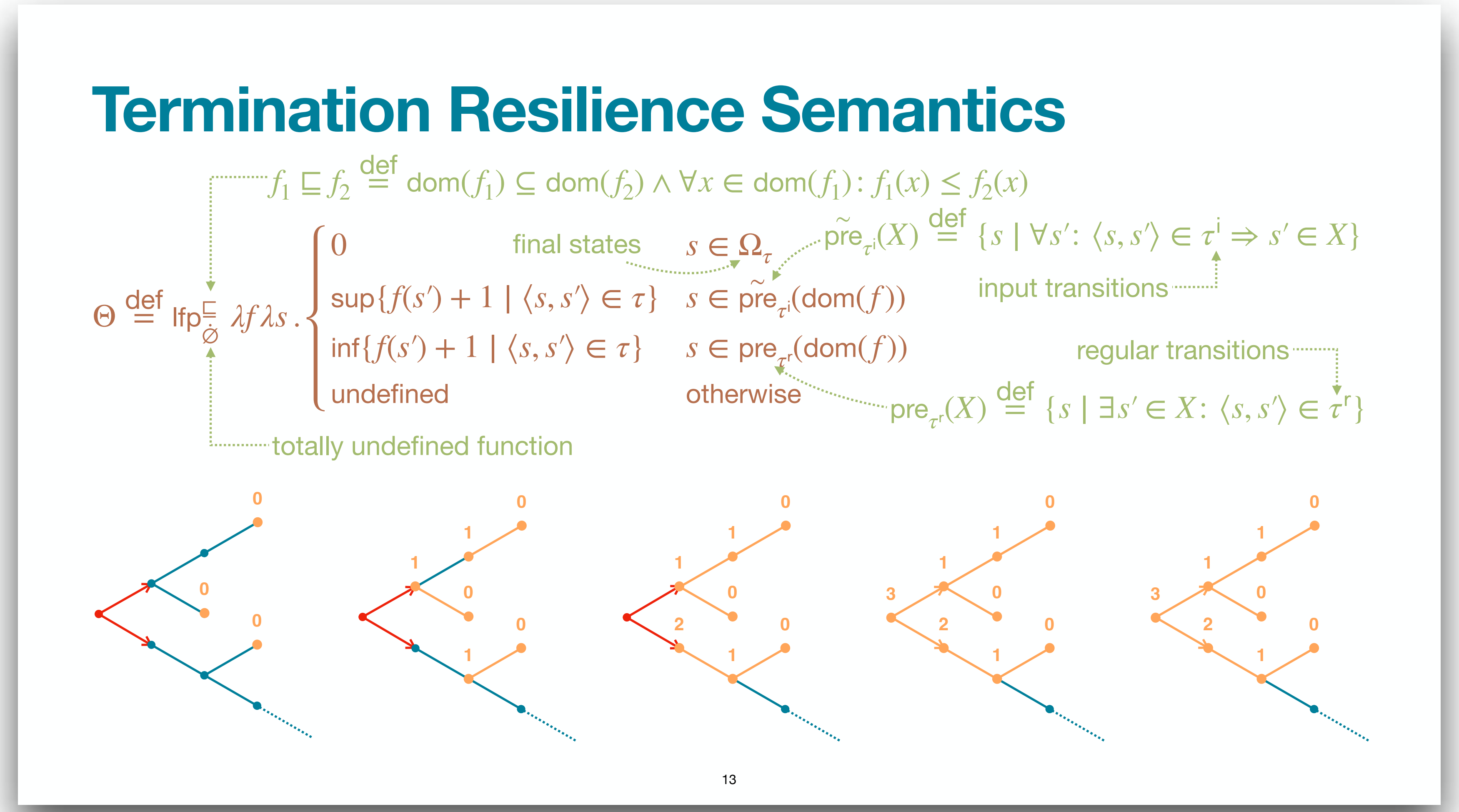
Termination Resilience Static Analysis

Loops

```
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
```



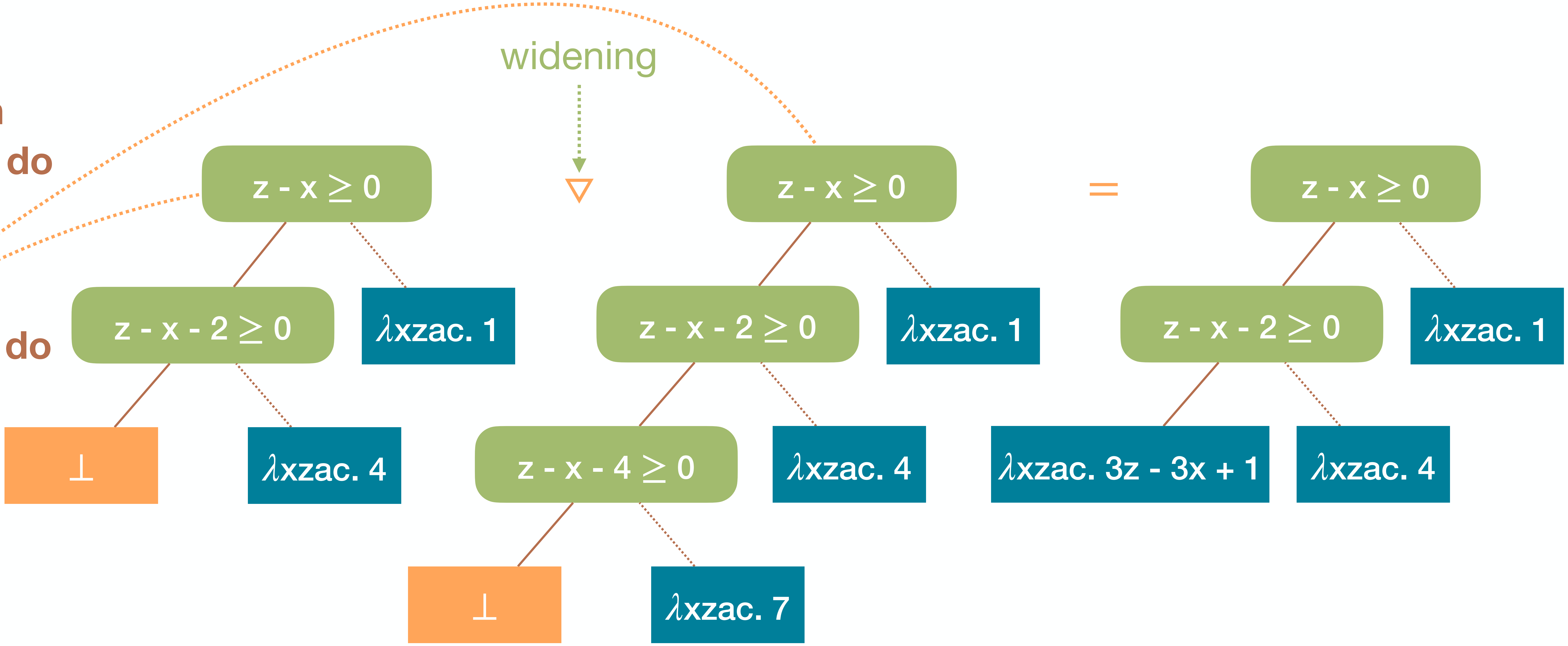




Termination Resilience Static Analysis

Loops

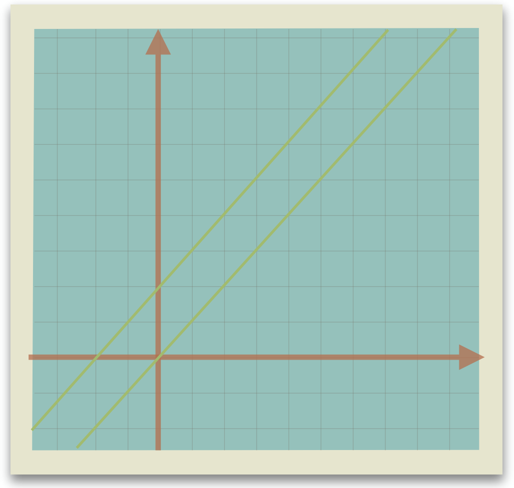
```
function f(x) {

  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
```

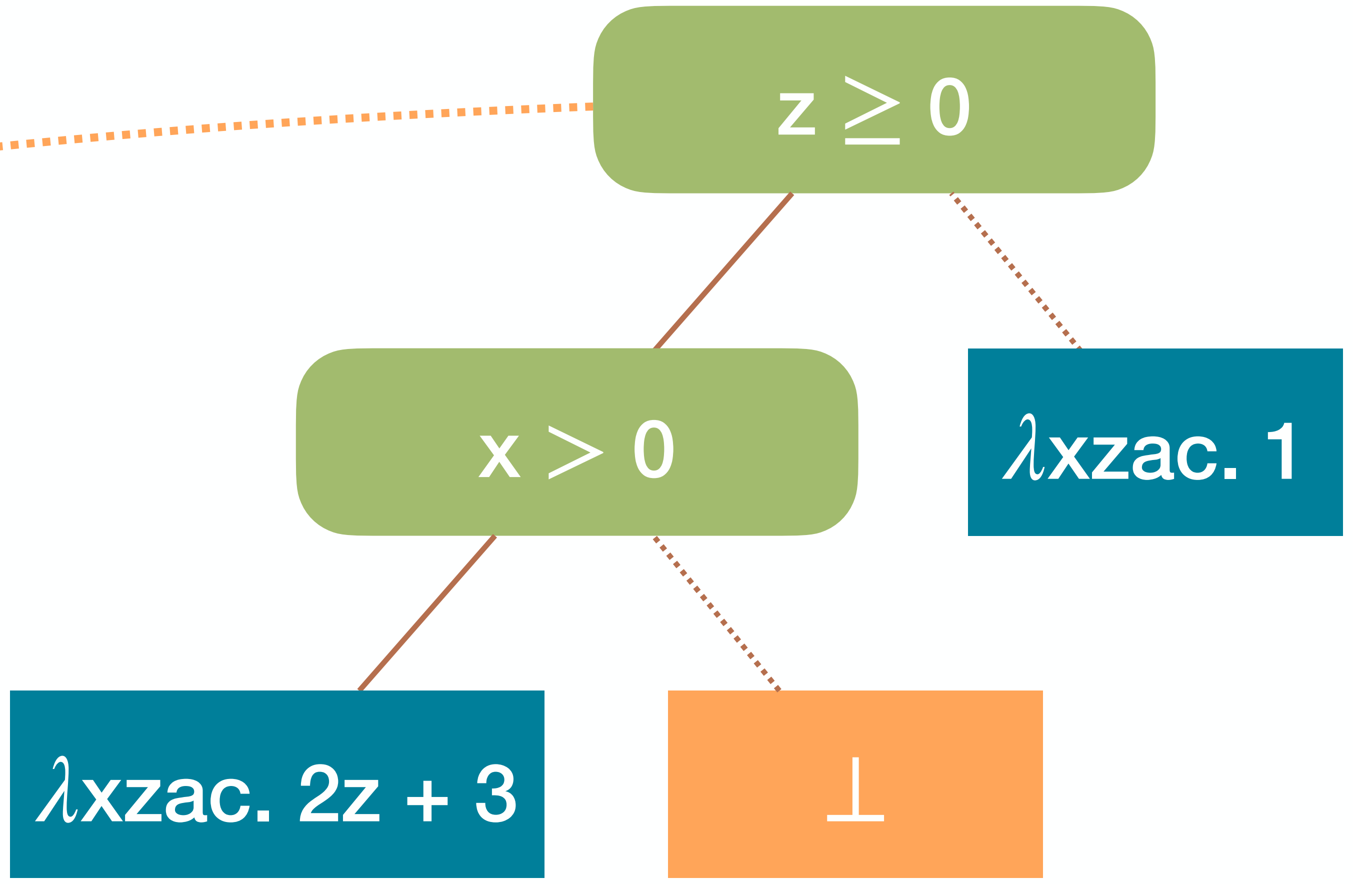


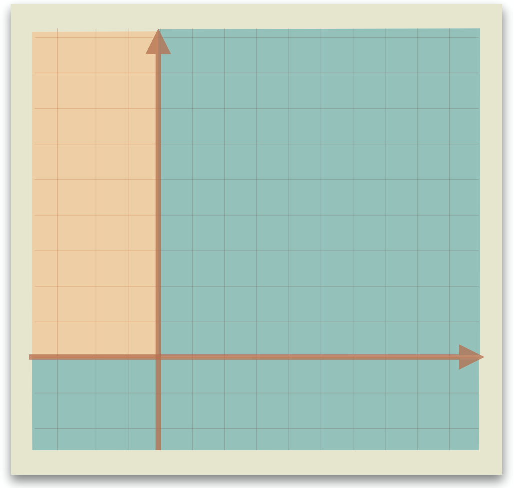
Termination Resilience Static Analysis

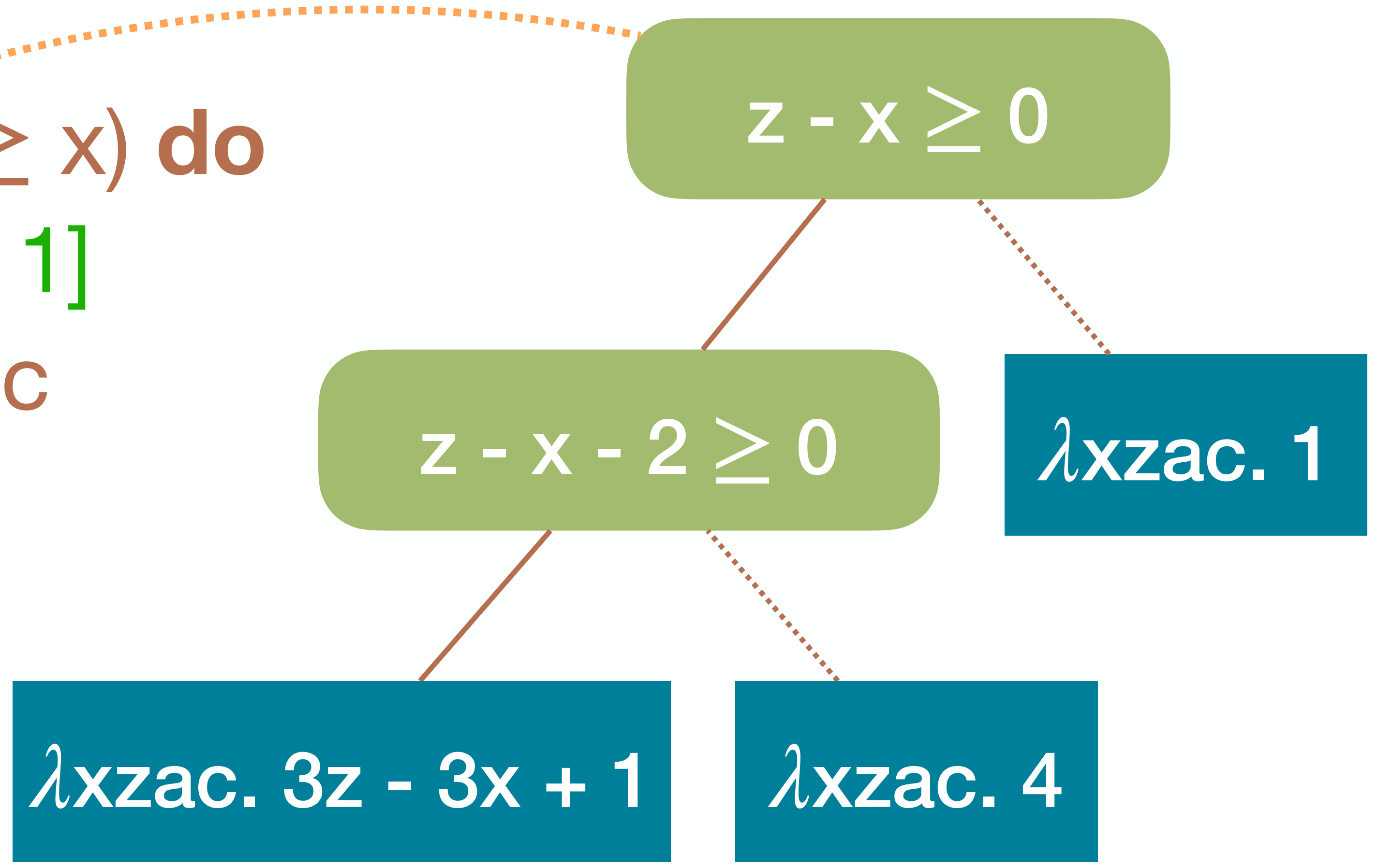
Loops

```
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
```

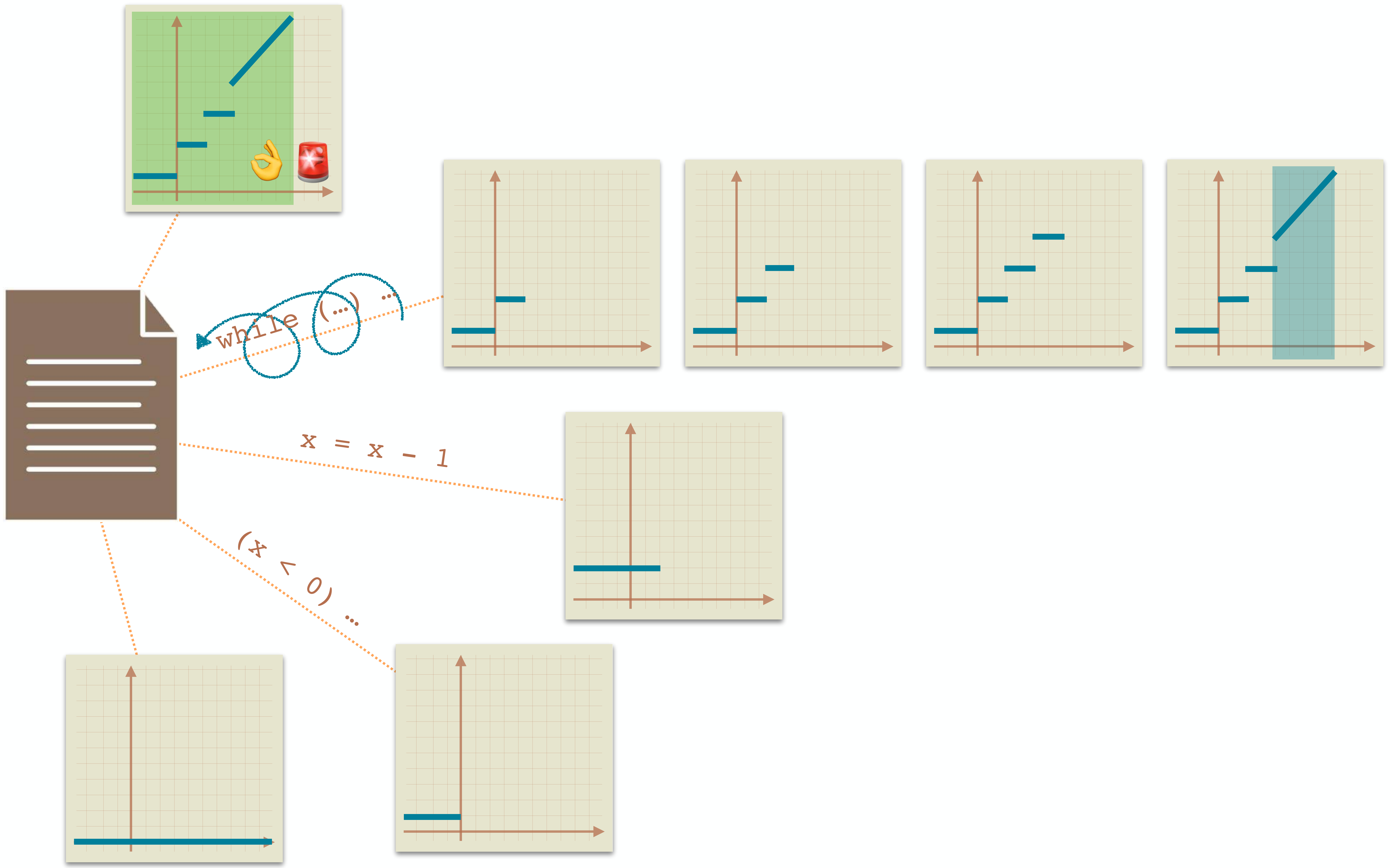








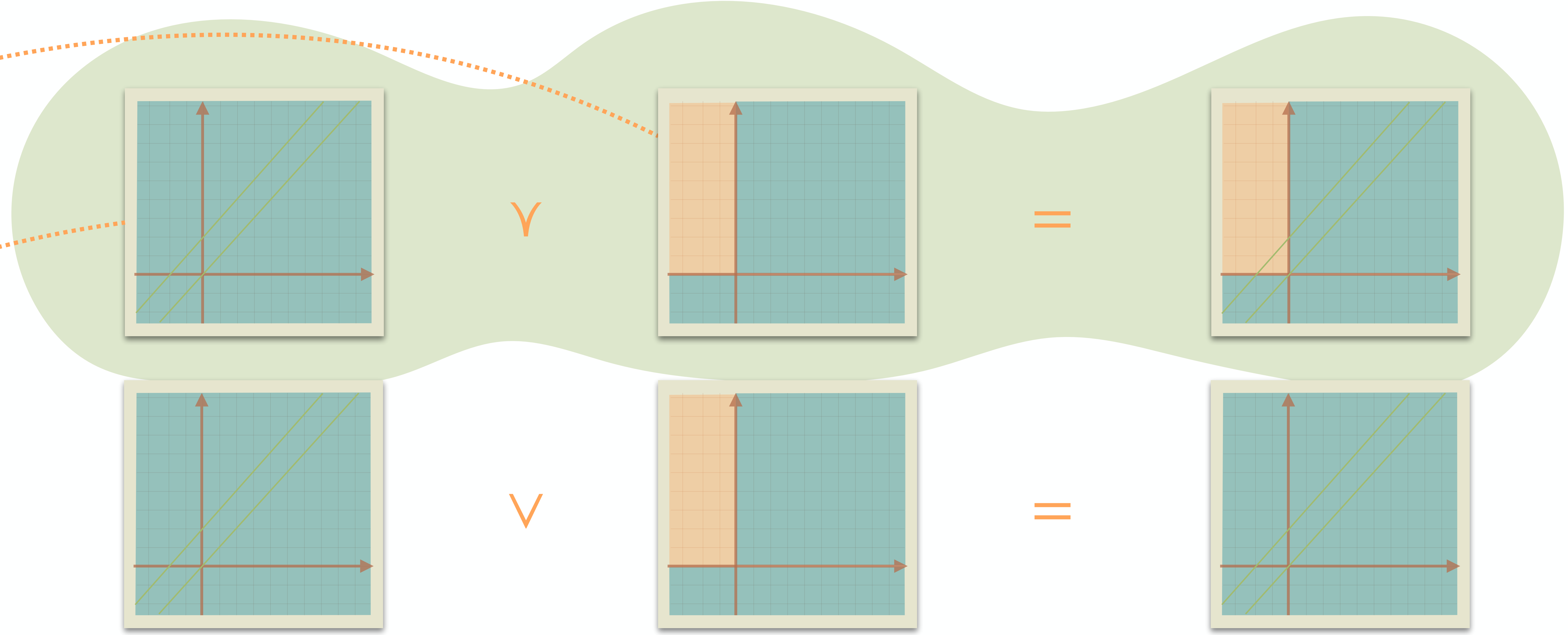
Termination Resilience Static Analysis



Termination Resilience Static Analysis

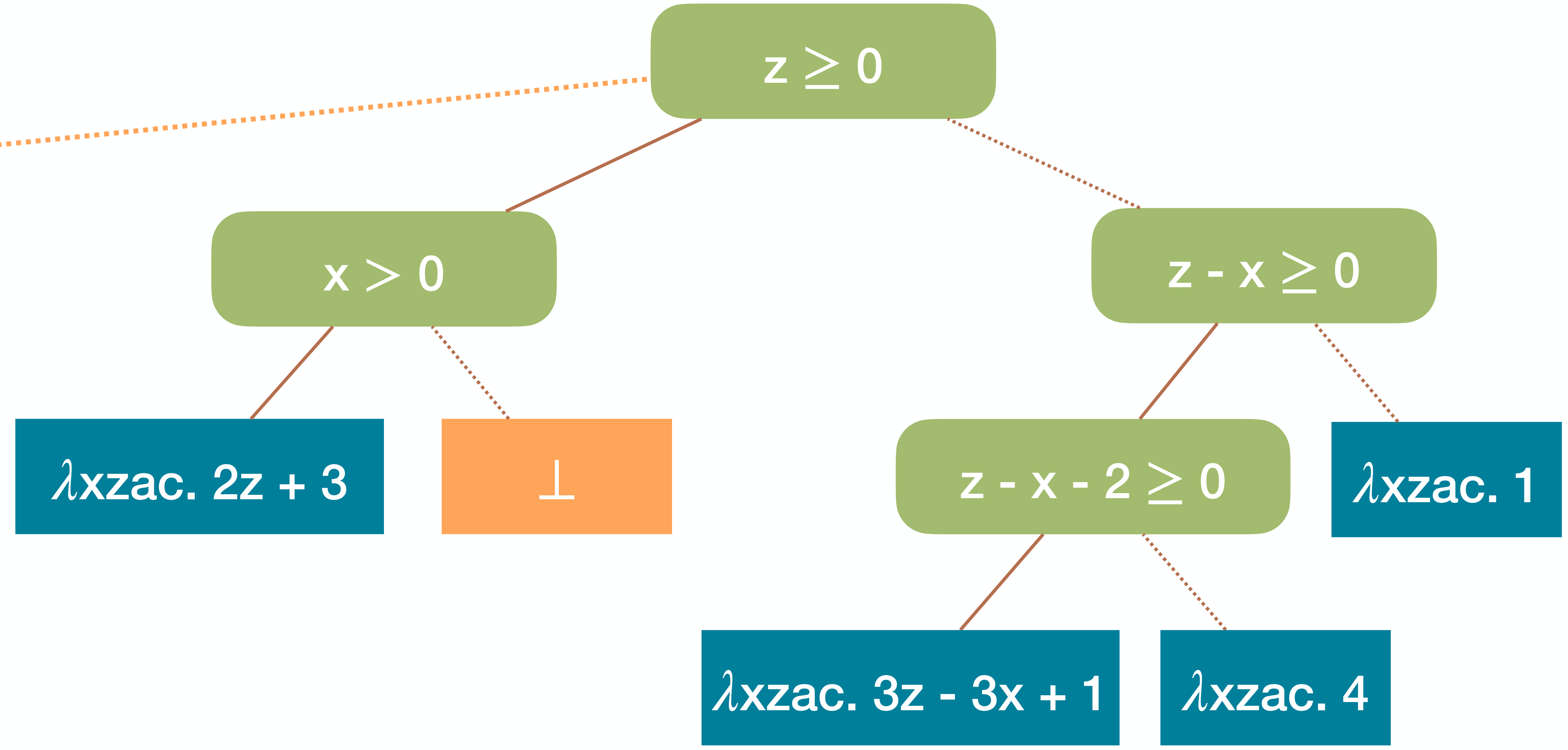
Approximation Join or Resilience Join?

```
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
```



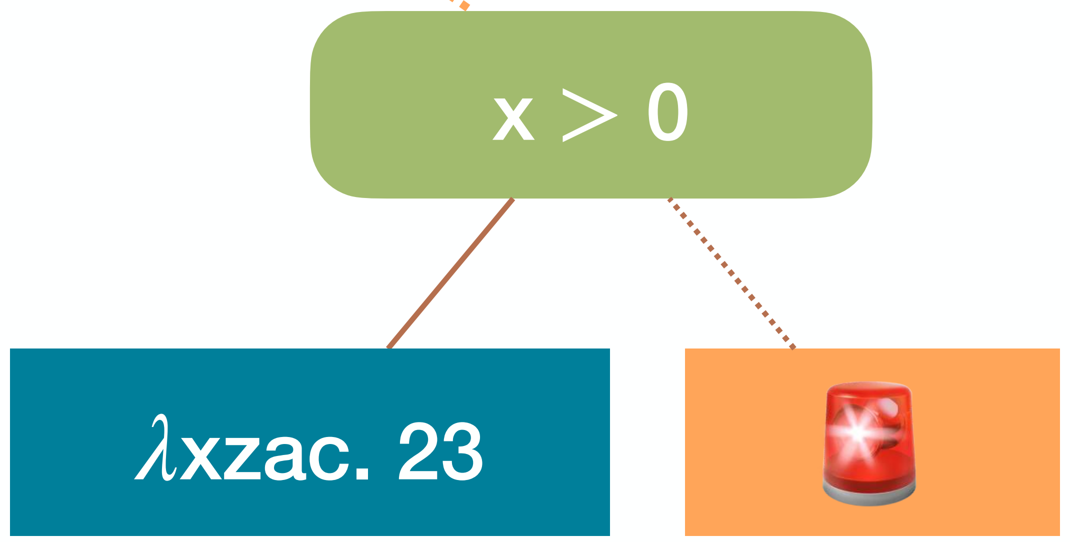
Termination Resilience Static Analysis

```
function f(x) {

  [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

```
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
```

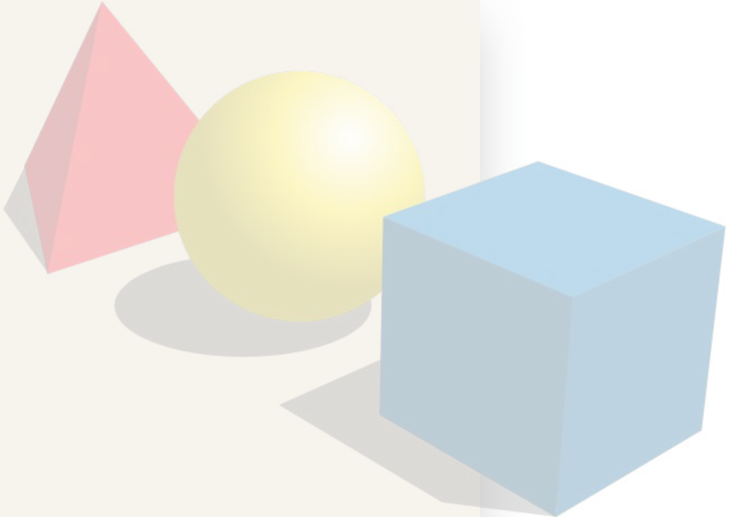


Termination Resilience Static Analysis

3-Step Recipe

practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties



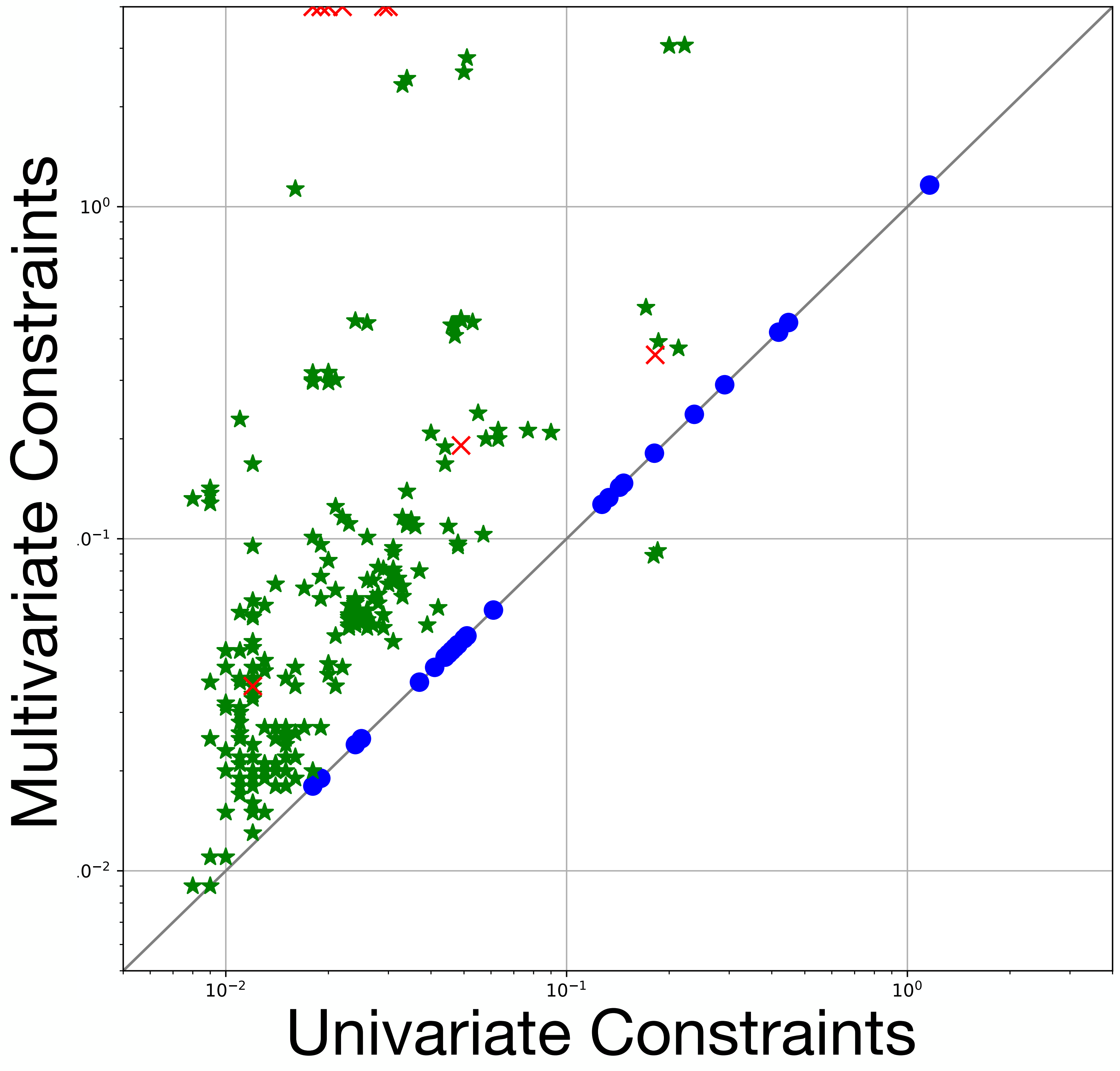
concrete semantics
mathematical models of the program behavior

Experimental Evaluation

	Benchmark	Property	Verified	Alarms	TO	Time
Univariate Constraints	SV-COMP 2024	Termination	0	119	0	3.5s
		Termination Resilience	61	58	0	3.6s
	Raad et al @ OOPSLA 2024	Termination	0	36	0	0.5s
		Termination Resilience	16	20	0	0.5s
	Shi et al. @ FSE 2022	Termination	0	85	0	2.0s
		Termination Resilience	57	28	0	2.2s
Multivariate Constraints	Benchmark	Property	Verified	Alarms	TO	Time
	SV-COMP 2024	Termination	0	119	0	7.2s
		Termination Resilience	76	43	0	16.9s
	Raad et al @ OOPSLA 2024	Termination	0	36	0	7.2s
		Termination Resilience	16	20	0	16.9s
	Shi et al. @ FSE 2022	Termination	0	85	0	69s
		Termination Resilience	49	28	8	500s

Experimental Evaluation

Univariate vs Multivariate Constraints



★ equal precision
× multivariate constraints are more precise
● univariate constraints are more precise (!)

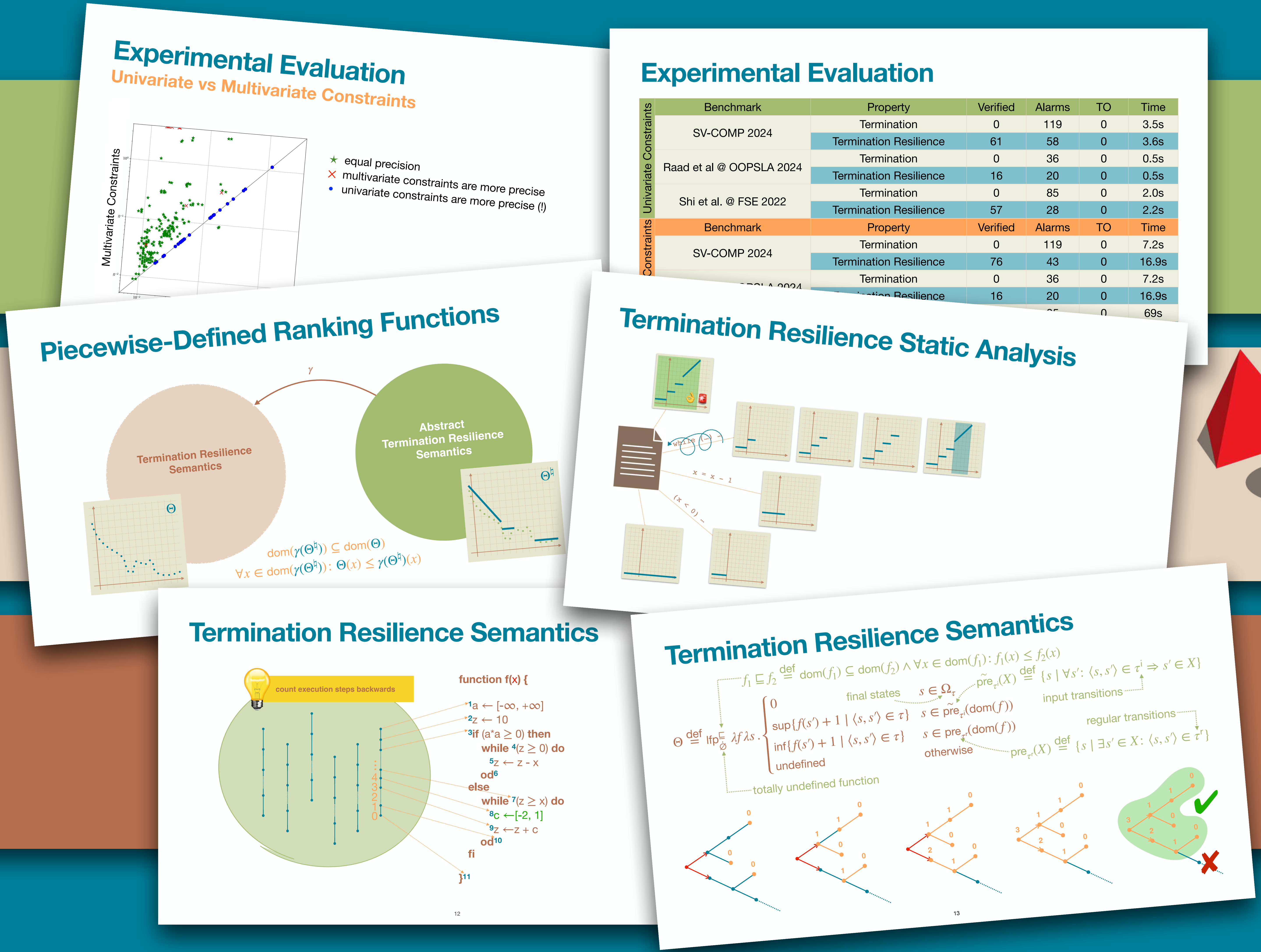
Termination Resilience Static Analysis

3-Step Recipe

practical tools

abstract semantics
abstract domains

concrete semantics



THANKS!