what is the difference between:

1. invariant
2. inductive invariant

$\{x \geq 0 \ \wedge \ x \text{ is even}\}$

```
while (x > 0)
    x = x - 2
    if x is odd
        x = -100
```

invariant $\boxed{x \geq 0}$

but it's not inductive

$\{x \geq 0\}$

$\{x \geq 0\}$ loop body $\{x \geq 0\}$ ✗

predicate abstraction

$\{pre\}$ stmt $\{true\}$

$\{x > 100 \wedge y = 0\}$

$x := x + 1$

$\{x > 100 \wedge y = 0\}$

Set of predicates $= \{x > 100, y = 0\}$

$\{x > 100 \wedge y = 0\}$ $x := x - 1$ $\{y = 0\}$	$\{\ldots\}$ $y = 1$ $\{x > 100 \wedge y \neq 0\}$

$\{x > 100 \wedge y = 0\}$

$x := -10$

$\{x \leq 100 \wedge y = 0\}$

Back to Horn clauses

$\{x=0 \wedge y=0\}$ $\qquad \{(x,y) \mid x,y \in \mathbb{Z}\}$

$\{x=y\}$

while $(n>0)$

$\quad x := x+1$

$\quad y := y+1$

$\quad n := n-1$

$\{x=y\}$

$\{x=10 \Rightarrow y=10\}$

---

initiation

$$x=0 \wedge y=0 \Longrightarrow \overset{x=y}{I(x,y)}$$

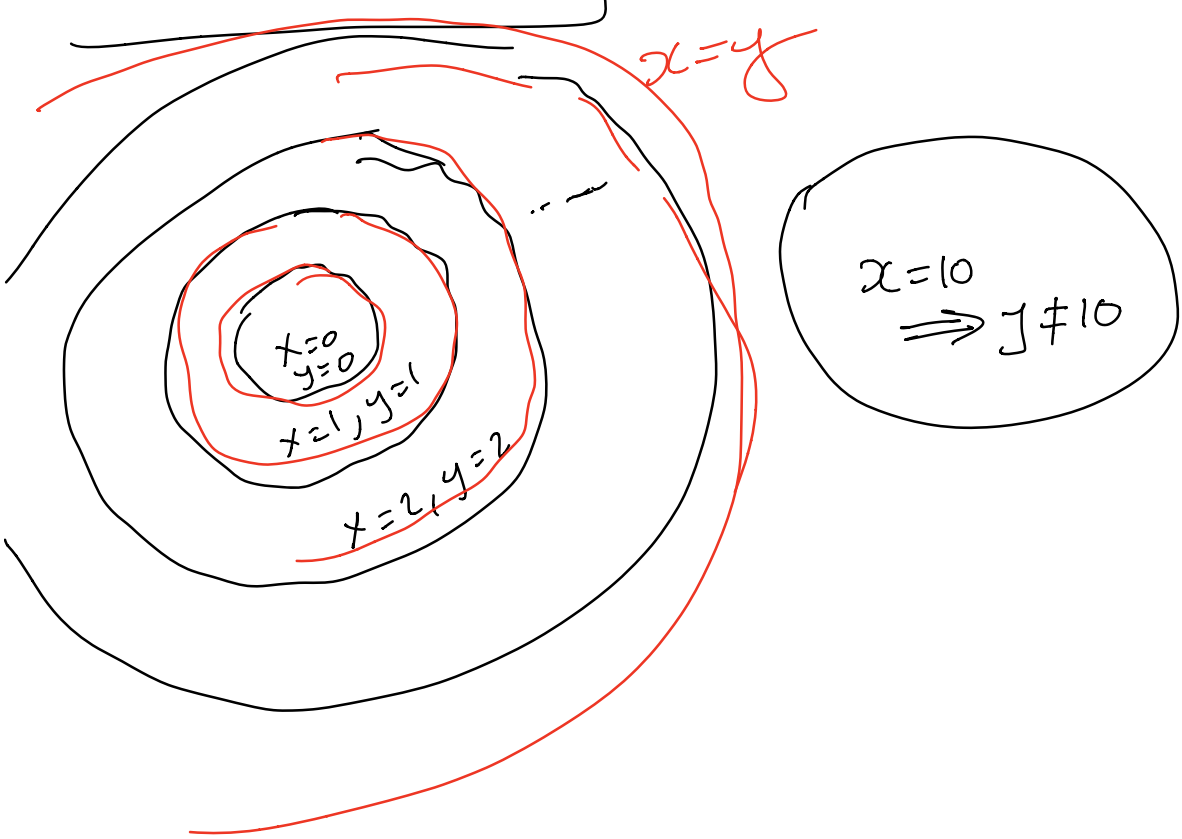
consecution

$$\overset{x=y}{I(x,y)} \wedge \underbrace{x'=x+1 \wedge y'=y+1}_{\text{Body of loop}} \Longrightarrow \overset{x'=y'}{I(x',y')}$$

Safety

$$\overset{x=y}{I(x,y)} \Longrightarrow (x=10 \Rightarrow y=10)$$

Visualize invariants



initially $I(x,y) = pre$

while not fixpoint

union

$$I(x',y') = I(x',y') \vee$$

$$\alpha_C\left(\underbrace{\exists x,y.}\ I(x,y) \wedge x' = x+1 \wedge y' = y+1\right)$$

we don't care about the states at the beginning of the loop

computes the least fixpoint

"smallest invariant"

Check that $I(x,y) \Longrightarrow \varphi_{post}$ is VALID

E.g.:

$I = x=0 \wedge y=0$

$I = x=0 \wedge y=0 \vee (x=1 \wedge y=1)$

$I = x=0 \wedge y=0 \vee (x=1 \wedge y=1) \vee (x=2 \wedge y=2$

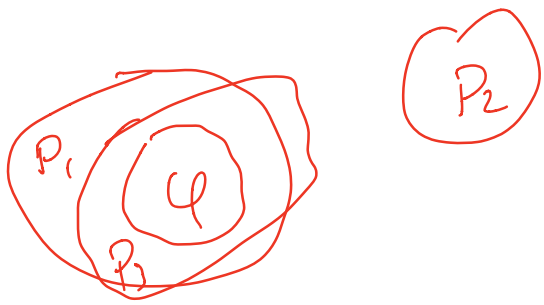
$\vdots$

predicate abstraction

Cartesian

$$Preds = \{P_1, \dots, P_n\}$$

given φ, what is the strongest formula ψ over Preds s.t. $\varphi \Rightarrow \psi$



$$\alpha_c(\varphi) = \bigwedge \{P_i \mid \varphi \Rightarrow P_i \text{ is VALID}\} \wedge$$

$$\bigwedge \{\neg P_i \mid \varphi \Rightarrow \neg P_i \text{ is VALID}\}$$

E.g. $\{x > 100 \wedge y = 0\}\ x := x - 1\ \{y = 0\}$

$$Preds = \{\underset{\times}{x' > 100},\ y' = 0\}$$

$$\alpha_c(\underbrace{x > 100 \wedge y = 0 \wedge y' = 0 \wedge x' = x - 1}) = y' = 0$$

Boolean abstraction

$\alpha_B(\varphi) =$

Let X be the set of all formulas of the form

$(\neg) p_1 \wedge (\neg) p_2 \wedge \cdots \wedge (\neg) p_n$

return

$$\bigvee \{ \phi \mid \phi \wedge \varphi \text{ is SAT}, \ \phi \in X \}$$

2^n elements