

Hoare Triple

$$\{Q\} S \{R\}$$

$\equiv$ Program S , when executed satisfying state Q , will terminate in state R

In ACSL Format:

```
//@ assert Q;  
S;  
//@ assert R;
```

Strength

$P \Rightarrow Q$: " P is stronger than Q "



"anything Q is, P is but stronger."

WP : Predicate transformer

$$\{Q\} S \{R\} \equiv Q \Leftarrow wp(S, R)$$

$$= wp(S, R)$$

- All states such that when program S is executed, it will terminate in state R

For any program S : $wp S$: Precondition $\rightarrow$ predicates

Notation: $wp(S, R) \equiv wp S R$

Axioms of Weakest Preconditions

0. Law of the Excluded Miracle:

$$wp\ S\ False = False$$

- the weakest precondition of a program to ensure False (as termination state) is just: False

1. Dist. over $\wedge$: $wp\ S\ (P \wedge R) \equiv wp\ S\ P \wedge wp\ S\ R$

2. Dist. over $\vee$: $wp\ S\ (P \vee R) \Leftarrow wp\ S\ P \vee wp\ S\ R$
- with equality if S is deterministic

3. Monotonicity: $(Q \Rightarrow R) \Rightarrow wp\ S\ Q \Rightarrow wp\ S\ R$
- it is order-preserving

4. Skip Rule: $wp\ skip\ R = R$
For example, $A; skip = A$ (Identity)
 $skip; A = A$

5. Abort Rule $wp\ Abort\ R = False$
- Remember: $F \rightarrow p$ For any p
OR $\emptyset = \{x \mid False\} \leq \{x \mid P\}$

6. Sequence: $wp\ (S; T)\ R = wp\ S\ (wp\ T\ R)$
- That is, $wp\ (S; T) = wp\ T; wp\ S$

ex: $(F \circ g)(x) = F(g(x)) \rightarrow wp\ S \circ wp\ T$
 $(g \circ F)(x) = F(g(x))$
"after" $\nwarrow$
"then" $\nearrow$

7. Definition: $S \approx T \equiv \forall R. wp\ S\ R \equiv wp\ T\ R$

For example: $x := 1; x := 5$ vs $x := 5$

- both have exact same side effects, they are equivalent

8. Theorem: $(A; B); C \approx A; (B; C)$
 - Follows from axiom 6

Grammar For Terms

Term ::= Const | Variable | Function application
 $t \quad \quad \quad v \quad \quad \quad x \quad \quad \quad f(t_0, t_1, \dots, t_n)$

Textual Substitution

" $t[\vec{x} := \vec{E}]$ " is the term obtained from " t " by simultaneously replacing vars $\vec{x}$ with terms $\vec{E}$

$\rightarrow c[x := E] = c$ if c Constant

$\rightarrow y[x := E] =$ if x and y are same variable:
 then E
 else y
 F:

$f(t_0, t_1, \dots, t_n)[x := E] = f(t_0[x := E], \dots, t_n[x := E])$

Example:

$(x + y - z \times z)[x, y := x + y, 0]$

$(x + y)[\dots] - (z \times z)[\dots]$

$x[\dots] + y[\dots] - z[\dots] \times z[\dots]$

$x + y + 0 - z \times z$

Note: $t[x, y := E, F] \neq (t[x := E])[y := F]$

* Assignment Rule: $wp(x := E) R = R[x := E]$ $\rightarrow$ $\text{//@ assert } x = E$
 $\hookrightarrow E$ must be well defined $\text{//@ assert } R$