

Programming Languages: Guarded Command Language Summary

Weakest Precondition

The weakest precondition transformer wp satisfies the following rules:

- $wp\ S\ False = False.$
- $wp\ S\ Q \wedge wp\ S\ R = wp\ S\ (Q \wedge R).$
- $wp\ S\ Q \vee wp\ S\ R \Rightarrow wp\ S\ (Q \vee R).$

Denote **if** $B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1$ **fi** by IF , **do** $B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1$ **od** by DO , and $B_0 \vee B_1$ by BB .

$$\begin{aligned}
 wp\ abort & \quad P = False \\
 wp\ skip & \quad P = P \\
 wp\ (x := E) & \quad P = P[x \backslash E] \\
 wp\ (S; T) & \quad P = wp\ S\ (wp\ T\ P) \\
 wp\ IF\ P & \\
 & = (B_0 \Rightarrow wp\ S_0\ P) \wedge (B_1 \Rightarrow wp\ S_1\ P) \wedge BB \\
 & = ((B_0 \wedge wp\ S_0\ P) \vee (B_1 \wedge wp\ S_1\ P)) \\
 wp\ DO\ P & = \\
 & \mu\ (\lambda X \rightarrow wp\ IF\ X \vee (\neg BB \wedge P)) ,
 \end{aligned}$$

where $\mu\ F$ denotes the strongest X such that $X = F\ X$. The $x := E$ case shall have a side condition that E is defined.

General case: denote by $B_i \rightarrow S_i$ the guarded commands $B_0 \rightarrow S_0 \mid \dots \mid B_{n-1} \rightarrow S_{n-1}$.

$$\begin{aligned}
 wp\ (\text{if } B_i \rightarrow S_i \text{ fi})\ P & = \\
 & \langle \forall i : 0 \leq i < n : B_i \Rightarrow wp\ S_i\ P \rangle \wedge \\
 & \langle \exists i : 0 \leq i < n : B_i \rangle .
 \end{aligned}$$

When $n = 0$, we have **if fi** = *abort*. Similarly,

$$\begin{aligned}
 wp\ (\text{do } B_i \rightarrow S_i \text{ od})\ P & = \mu\ (\lambda X \rightarrow \\
 & wp\ (\text{if } B_i \rightarrow S_i \text{ fi})\ X \vee \\
 & (\neg \langle \exists i : 0 \leq i < n : B_i \rangle \wedge P)) .
 \end{aligned}$$

When $n = 0$, we have **do od** = *skip*.

Hoare Logic

Definition: $\{P\} S \{Q\} \equiv P \Rightarrow wp\ S\ Q.$

The definition entails that

$$\begin{aligned}
 \{P\} skip \{Q\} & \equiv P \Rightarrow Q \\
 \{P\} x := E \{Q\} & \equiv P \Rightarrow Q[x \backslash E] \\
 \{P\} S; T \{Q\} & \equiv \\
 & \langle \exists R :: \{P\} S \{R\} \wedge \{R\} T \{Q\} \rangle \\
 \{P\} IF \{Q\} & \equiv \\
 & (P \Rightarrow B_0 \vee B_1) \wedge \\
 & \{P \wedge B_0\} S_0 \{Q\} \wedge \{P \wedge B_1\} S_1 \{Q\}
 \end{aligned}$$

There is also a side condition that E is defined.

Regarding loops, by the *Fundamental Invariance Theorem*, the weakest precondition of DO entails that $\{P\} DO \{Q\}$ follows from

1. $P \wedge \neg B_0 \wedge \neg B_1 \Rightarrow Q,$
2. $\{P \wedge B_0\} S_0 \{P\}$ and $\{P \wedge B_1\} S_1 \{P\},$ and
3. there exists an integer function t on the state space such that
 - (a) $[P \wedge (B_0 \vee B_1) \Rightarrow t \geq 0],$
 - (b) $\{P \wedge B_0 \wedge t = C\} S_0 \{t < C\},$ and
 - (c) $\{P \wedge B_1 \wedge t = C\} S_1 \{t < C\}.$

Properties Hoare triples satisfy the following rules:

- $\{P\} S \{false\} \equiv \neg P,$
- $\{P\} S \{Q\} \wedge (P_0 \Rightarrow P) \Rightarrow \{P_0\} S \{Q\},$
- $\{P\} S \{Q\} \wedge (Q \Rightarrow Q_0) \Rightarrow \{P\} S \{Q_0\},$
- $\{P\} S \{Q\} \wedge \{P\} S \{R\} \Rightarrow \{P\} S \{Q \wedge R\},$
- $\{P\} S \{Q\} \wedge \{R\} S \{Q\} \Rightarrow \{P \vee R\} S \{Q\}.$

References

- [Dij76] E. W. Dijkstra. *A Discipline of Programming*. Prentice Hall, 1976.
- [Kal90] A. Kaldewaij. *Programming: the Derivation of Algorithms*. Prentice Hall, 1990.