Chapter 2: The Guarded Command Language (Part 2)

Skip

- Execution of skip does not have any effect.

$$\{P\}\text{skip}\{Q\} \text{ is equivalent to } [P \Rightarrow Q]$$

- Example

$$
\begin{array}{l}
|[\\
\textbf{var } x, y : \text{int}; \\
\{x \geq 1\} \\
\text{skip} \\
\{x \geq 0\} \\
]|
\end{array}
$$

- Weakest precondition: $wp.\text{skip}.Q \equiv Q$

Assignment

- Any change of state is due to the execution of an assignment statement.

$$x := E$$

replaces the value of x by the value of E.

$$\{P\}x := E\{Q\} \text{ is equivalent to } [P \Rightarrow \text{def}.E \wedge Q(x := E)]$$

Here def.E is defined for which values of its variables in E is defined.

$$\text{def}.(a \;\mathbf{mod}\; b) = b \neq 0$$

- Weakest precondition

$$[wp.(x := E).Q \;\equiv\; \mathbf{def}.E \;\wedge\; Q(x := E)]$$

- Example

$$\{x \geq 3\} x := x + 1 \{x \geq 0\}$$

follows from

$$
\begin{array}{ll}
 & \text{def.}(x+1) \wedge (x \geq 0)(x := x + 1) \\
\equiv & \quad \{ \text{ def } \} \\
 & \mathit{true} \wedge (x \geq 0)(x := x + 1) \\
\equiv & \quad \{ \wedge\text{, substitution } \} \\
 & x + 1 \geq 0 \\
\equiv & \quad \{ \text{ arithmetic } \} \\
 & x \geq -1 \\
\Leftarrow & \quad \{ \text{ arithmetic } \} \\
 & x \geq 3
\end{array}
$$

Catenation

- Catenation allows us to describe sequence of actions.

$$S; T$$

S is executed after which T is executed.

$$\{P\}S; T\{Q\} \text{ is equivalent to } \exists R.\{P\}S\{R\} \text{ and } \{R\}T\{Q\}$$

- Weakest precondition

$$[wp.(S; T).Q \;\equiv\; wp.S.(wp.T.Q)]$$

i.e., semi-colon corresponds to function composition.

- Prove

$$
\begin{array}{l}
|[\\
\textbf{var } a, b : \text{bool}; \\
\{(a \equiv A) \wedge (b \equiv B)\} \\
a := a \equiv b; \\
b := a \equiv b; \\
a := a \equiv b \\
\{(a \equiv B) \wedge (b \equiv A)\} \\
]|
\end{array}
$$

  - Hint: Compute the weakest preconditions backwards.

Selection

$$\textbf{if } B.0 \rightarrow S.0 \ [] \ \cdots \ [] \ B.n \rightarrow S.n \textbf{ fi}$$

where

- $B.i$: a boolean expression (a guard)
- $S.i$: a statement
- $B.i \rightarrow S.i$: a guarded command

1. All guards B_i are evaluated.
2. If none of the guards evaluates to true then execution *aborts*, otherwise one of the guards that has the value true is chosen *non-deterministically* and the corresponding statement is executed.

An Example

Derive a statement S that satisfies

$$
\begin{array}{l}
|[\\
\mathbf{var}\ x, y, z : \mathrm{int}; \\
\{\mathrm{true}\} \\
S \\
\{z = x\ \mathbf{max}\ y\} \\
]|
\end{array}
$$

where **max** is defined by

$$z = x\ \mathbf{max}\ y\ \equiv\ (z = x \vee z = y) \wedge z \geq x \wedge z \geq y$$

We conclude that $z := x$ is a candidate for S. As a precondition we can have

$$
\begin{array}{ll}
 & ((z = x \vee z = y) \wedge z \geq x \wedge z \geq y)(z := x) \\
\equiv & \qquad \{ \text{ substitution } \} \\
 & (x = x \vee x = y) \wedge x \geq x \wedge x \geq y \\
\equiv & \qquad \{ \text{ calculus } \} \\
 & x \geq y
\end{array}
$$

So

$$x \geq y \rightarrow z := x$$

Symmetrically,

$$y \geq x \rightarrow z := y$$

So, the definition of S is

$$\textbf{if } x \geq y \rightarrow z := x \ [] \ y \geq x \rightarrow z := y \textbf{ fi}$$

Formulation of Selection Statement

$$\{P\}\textbf{if } B_0 \rightarrow S_0 \;[]\; B_1 \rightarrow S_1 \textbf{ fi}\{Q\}$$

is equivalent to

1. $[P \rightarrow B_0 \vee B_1]$ and
2. $\{P \wedge B_0\}S_0\{Q\}$ and $\{P \wedge B_1\}S_1\{Q\}$

- Examples
  - Prove $\{x = 0\}\textbf{if } \mathit{true} \rightarrow x := x + 1 \;[]\; \mathit{true} \rightarrow x := x + 1 \textbf{ fi}\{x = 1\}$.
  - Prove $\{x = 0\}\textbf{if } \mathit{true} \rightarrow x := 1 \;[]\; \mathit{true} \rightarrow x := -1 \textbf{ fi}\{x = 1 \vee x = -1\}$.

Weakest Precondition for Selection

$$
\begin{aligned}
&[wp.(\textbf{if } B_0 \rightarrow S_0 \ [] \ B_1 \rightarrow S_1 \ \textbf{fi}).Q \\
&\quad \equiv (B_0 \vee B_1) \ \wedge \\
&\qquad (B_0 \rightarrow wp.S_0.Q) \ \wedge \\
&\qquad (B_1 \rightarrow wp.S_1.Q) \\
&]
\end{aligned}
$$

Repetition

$$\textbf{do } B.0 \rightarrow S.0 \;[]\; \cdots \;[]\; B.n \rightarrow S.n \textbf{ od}$$

1. All guards B_i are evaluated.
2. If none of the guards evaluates to true then execution *skip*, otherwise one of the guards that has the value true is chosen *non-deterministically* and the corresponding statement is executed, after which the repetition is executed again.

Formulation of Repetition Statement

$$\{P\}\textbf{do } B_0 \rightarrow S_0 \ [] \ B_1 \rightarrow S_1 \textbf{ od}\{Q\}$$

is equivalent to

$$\begin{array}{l}
\{P\} \\
\textbf{if } (\neg B_0 \wedge \neg B_1) \rightarrow \text{skip} \\
\ [] \ B_0 \rightarrow S_0; \ \textbf{do } B_0 \rightarrow S_0 \ [] \ B_1 \rightarrow S_1 \textbf{ od} \\
\ [] \ B_1 \rightarrow S_1; \ \textbf{do } B_0 \rightarrow S_0 \ [] \ B_1 \rightarrow S_1 \textbf{ od} \\
\textbf{fi} \\
\{Q\}
\end{array}$$

So

(i) $[P \wedge (\neg B_0 \wedge \neg B_1) \Rightarrow Q]$ and

(ii) $\{P \wedge B_0\}S_0\{P\}$ and $\{P \wedge B_1\}S_1\{P\}$

implies

$$\{P\}\textbf{do } B_0 \rightarrow S_0 \ [] \ B_1 \rightarrow S_1 \textbf{ od}\{Q\}$$

provided that this repetition terminates.

Note: A predicate P that satisfies (ii) is called an invariant of $\textbf{do } B_0 \rightarrow S_0 \ [] \ B_1 \rightarrow S_1 \textbf{ od}$.

An Example

Prove that

$$
\begin{array}{l}
|[\\
\textbf{var } x, y : \text{int}; \\
\{x = X \wedge y = Y \wedge x > 0 \wedge y > 0\} \\
\textbf{do } x > y \rightarrow x := x - y \;[]\; y > x \rightarrow y := y - x \textbf{ od} \\
\{x = X \textbf{ gcd } Y\} \\
]|
\end{array}
$$

Here, X **gcd** Y denotes the greatest common divisor of X and Y, and **gcd** has the following properties:

$$
\begin{array}{l}
x \textbf{ gcd } x = x \\
x \textbf{ gcd } y = y \textbf{ gcd } x \\
x > y \Rightarrow x \textbf{ gcd } y = (x - y) \textbf{ gcd } y \\
y > x \Rightarrow x \textbf{ gcd } y = x \textbf{ gcd } (y - x)
\end{array}
$$

The point is to define a suitable invariant P.

$$
\begin{array}{l}
|[\\
\mathbf{var}\ x, y : \mathrm{int}; \\
\{x = X \wedge y = Y \wedge x > 0 \wedge y > 0\} \\
\{P\} \\
\mathbf{do}\ x > y \rightarrow x := x - y \ [] \ y > x \rightarrow y := y - x\ \mathbf{od} \\
\{x = X\ \mathbf{gcd}\ Y\} \\
]|
\end{array}
$$

Proof Sketch.

1. Define an invariant P as

$$P : \; x > 0 \wedge y > 0 \wedge x \,\mathbf{gcd}\, y = X \,\mathbf{gcd}\, Y$$

   satisfying $x = X \wedge y = Y \wedge x > 0 \wedge y > 0 \;\Rightarrow P$.

2. Prove:
   - $P \wedge \neg(x > y) \wedge \neg(y > x) \;\Rightarrow\; x = X \,\mathbf{gcd}\, Y$
   - $\{P \wedge (x > y)\}x := x - y\{P\}$
   - $\{P \wedge (y > x)\}y := y - x\{P\}$

3. Show the termination of the repetition.

- Let $t = x + y$. $t \geq 0$ and t decreases in each step of repetition.
  - $P \wedge (x > y \vee y < x) \Rightarrow t \geq 0$
  - $\{P \wedge (x > y) \wedge t = C\} x := x - y \{t < C\}$
  - $\{P \wedge (y > x) \wedge t = C\} y := y - x \{t < C\}$

Constants

The following is not satisfactory:

$$
\begin{array}{l}
[[\\
\mathbf{var}\ A, B, x : int; \\
\{A > 0 \land B > 0\} \\
\quad gcd \\
\{x = A\ gcd\ B\} \\
]]
\end{array}
$$

as $A, B, x := 1, 1, 1$ is a possible solution.

Constant should not be changed.

$$
\begin{array}{l}
[\![\\
\textbf{con}\ A, B : int; \\
\textbf{var}\ x : int; \\
\{A > 0 \wedge B > 0\} \\
\quad gcd \\
\{x = A\ gcd\ B\} \\
]\!]
\end{array}
$$

Inner Blocks

$$
\begin{array}{l}
|[\\
\textbf{con } A, B : \mathit{int}; \{A > 0 \wedge B > 0\} \\
\textbf{var } x : \mathit{int}; \\
\quad |[\\
\quad \textbf{var } y : \mathit{int} \\
\quad x, y := A, B; \\
\quad \textbf{do} \\
\qquad x > 0 \rightarrow x := x - y \\
\qquad [] y > x \rightarrow y := y - x \\
\quad \textbf{od} \\
\quad \{x = A \textbf{ gcd } B \wedge y = A \textbf{ gcd } B\} \\
\quad]| \\
\quad \{x = A \textbf{ gcd } B\} \\
]|
\end{array}
$$

Used to extend the state (locally) by means of new variables.

$$\{P\}|[\mathbf{var}\ y\ ; S]|\{Q\}$$

is equivalent to

$$\{P\}S\{Q\}$$

provided that y does not occur in both P and Q.

Arrays

Arrays are used to represent a set of variables.

$$f : \textbf{array } [p..q) \textbf{ of } int$$

defines a program variable f which has as value a function:

$$[p..q) \rightarrow \mathcal{Z}.$$

Exercises

2.1 Prove

$$
\begin{array}{l}
|[\\
\textbf{var } x, y : int; \\
\{x = A \wedge y = B\} \\
x := x - y;\ y := x + y;\ x := y - x \\
\{x = B \wedge y = A\} \\
]|.
\end{array}
$$

2.2 Determine the weakest P such that

$$
\begin{array}{l}
|[\\
\mathbf{var}\ x : \mathit{int}; \\
\{P\} \\
x := x + 1; \\
\mathbf{if}\ x > 0\ \rightarrow\ x := x - 1 \\
\ [\,]\ x < 0 \rightarrow x := x + 2 \\
\ [\,]\ x = 1 \rightarrow \mathit{skip} \\
\mathbf{fi} \\
\{x \geq 1\} \\
]|.
\end{array}
$$

2.3 Prove the correctness of the following program.

$$
\begin{array}{l}
|[\\
\mathbf{var}\ x, y, z : int; \\
\{true\} \\
\mathbf{do}\ x < y \rightarrow x := x + 1 \\
\ [\!]\ y < z \rightarrow y := y + 1 \\
\ [\!]\ z < x \rightarrow z := z + 1 \\
\mathbf{od} \\
\{x = y = z\} \\
]|
\end{array}
$$

2.4 The following problem may be used to compute (non-deterministically) natural numbers x and y such that $x * y = N$. Prove:

$$
\begin{array}{l}
|[\\
\textbf{var } p, x, y, N : \text{int}; \\
\{N \geq 1\} \\
p, x, y := N - 1, 1, 1; \\
\{N = x * y + p\} \\
\textbf{do } p \neq 0 \\
\quad \rightarrow \textbf{if } p \textbf{ mod } x = 0 \rightarrow p, y := p - x, y + 1 \\
\qquad [\!]\ p \textbf{ mod } y = 0 \rightarrow x, p := x + 1, p - y \\
\qquad \textbf{fi} \\
\textbf{od} \\
\{x * y = N\} \\
]|.
\end{array}
$$

2.5 Prove

$$
\begin{array}{l}
|[\\
\mathbf{con}\ N : \mathit{int}\ \{N \geq 0\}; \\
f :\ \mathbf{array}\ [0..N)\ \mathbf{of}\ \mathit{int}; \\
\mathbf{var}\ b : \mathit{bool}; \\
\quad |[\\
\quad \mathbf{var}\ n : \mathit{int}; \\
\quad b, n := \mathit{false}, 0; \\
\quad \mathbf{do}\ n \neq N \rightarrow b := b \vee f.n = 0;\ n := n + 1\ \mathbf{od} \\
\quad]| \\
\{b \equiv (\exists i : 0 \leq i < N :\ f.i = 0)\} \\
]|.
\end{array}
$$