Hoare Logic

CS 81

September 30, 2010

State Transformers

State before		State after
	`x = x + 1`	
`x==1`		`x==2`
`x==7`		`x==8`
`even(x)`		`odd(x)`
`x > 3`		`x > 4`
precondition		*postcondition*

Preconditions and Postconditions

precondition → `x = x + 1` → *postcondition*

`x == 1`

`x > 10`

$\top$

$\bot$

Preconditions and Postconditions

precondition → `x = x + 1` → *postcondition*

`x == 1`

`x > 10`

$\top$

$\bot$

Design by Contract

precondition

postcondition

implementation

Specification for Sorting?

precondition

int a[N];

postcondition

$$\forall i.\ (1 \le i < \mathsf{N}) \rightarrow \mathtt{a}[i\text{-}1]\ \le\ \mathtt{a}[i]$$

implementation

```
for (int i=0; i<n; ++i)
   a[i] = 0;
```

Hoare Triples

$\{\ precondition\ \}$ `code` $\{\ postcondition\ \}$

Compare

$$\{ P \}\ c\ \{ Q \}$$

$$\{ P \}\ c\ \{ Q' \}$$

$$\{ P' \}\ c\ \{ Q \}$$

Assume $P \rightarrow P'$ and $Q \rightarrow Q'$

First Rule of Hoare Logic

$$\frac{P \to P' \qquad \{\ P'\ \}\ c\ \{\ Q'\ \} \qquad Q' \to Q}{\{\ P\ \}\ c\ \{\ Q\ \}}\ \text{IMPLIED}$$

Sequencing

$\{ P_1 \}\ c_1\ \{ Q_1 \}$

$\{ P_2 \}\ c_2\ \{ Q_2 \}$

$\{ ??? \}\ c_1 ; c_2\ \{ ??? \}$

Second Rule of Hoare Logic

$$\frac{\{\ P\ \}\ c_1\ \{\ R\ \} \qquad \{\ R\ \}\ c_2\ \{\ Q\ \}}{\{\ P\ \}\ c_1; c_2\ \{\ Q\ \}}\ \textsc{composition}$$

What if Our Conditions Don't Match?

$$\{\top\}\ \mathsf{c}_1\ \{x > 7\}$$

$$\{x > 5\}\ \mathsf{c}_2\ \{y = 2\}$$

$$\{???\}\ \mathsf{c}_1 ; \mathsf{c}_2\ \{???\}$$

If Statement

$$\frac{\{\ P \land e\ \}\ c_1\ \{\ Q\ \} \qquad \{\ P \land \neg e\ \}\ c_2\ \{\ Q\ \}}{\{\ P\ \}\ \texttt{if}\ (e)\ c_1\ \texttt{else}\ c_2\ \{\ Q\ \}}\ \mathrm{IF}$$

While Statement

$$\frac{\{\, I \wedge e \,\}\ c\ \{\, I \,\}}{\{\, I \,\}\ \texttt{while}\ (e)\ c\ \{\, I \wedge \neg e \,\}}\ \textsc{While}$$

Assignments

$$\frac{}{\{\ P[e/x]\ \}\ \ x\ \texttt{=}\ e\texttt{;}\ \ \{\ P\ \}}\ \textsc{Assignment}$$

Huth & Ryan: "Proof Tableaux"

$\{ P_1 \}$
c_1
$\{ P_2 \}$
c_2
$\{ P_3 \}$
...
$\{ P_n \}$
c_n
$\{ P_{n+1} \}$

Hint
Work Bottom Up!

Example

$\{ y = 5 \}$ `y = y + 1` $\{ y = 6 \}$

$\{ y = 5 \}$

$\{ y + 1 = 6 \}$ implied

`y = y + 1`

$\{ y = 6 \}$ assignment

Exercise: Swap

$\{ x = x_0 \wedge y = y_0 \}$ `t = x; x = y; y = t` $\{ x = y_0 \wedge y = x_0 \}$

Exercise: Max

$\{\top\}$ `if (x > y) m = x else m = y` $\{ m = \max(x,y) \}$

Exercise: While

$\{ x \leq n \}$ `while (x < n) x = x+1` $\{ x = n \}$

Note: *Partial* Correctness

$\{ \top \}$

```
while (true) {
    x = x+1;
}
```

$\{ y = 99 \}$

Total Correctness

=

Partial Correctness

+

Termination

Proving Termination

One approach: Define a *variant*

(non-negative but decreases on each iteration)

Total Correctness?

```
x = 0;
while ( n > 0 )
  {
    x = x+1;
    n = n-1;
  }
```

Termination?

```
n = 0;
while ( n < 99 )
   {
      x = x+1;
      n = n+1;
   }
```

Total Correctness?

$\{ m = m_0 > 0 \wedge n = n_0 > 0 \}$

```
while ( m != n ) {
  if ( m < n )
    n = n-m;
  else
    m = m-n;
}
```

$\{ m = \gcd(m_0, n_0) \}$

Total Correctness?

$\{ x = 0 \wedge y = 1 \wedge z = 1 \wedge n \geq 1 \}$

```
while ( z < n ) {
  y = x + y;
  x = y - x;
  z = z + 1;
}
```

$\{ y = fib(n) \}$