CS4004/CS4504: FORMAL VERIFICATION

Lectures 14: Hoare Logic

Vasileios Koutavas

School of Computer Science and Statistics
Trinity College Dublin

Prove the following Hoare triples:

- $(\!|y > 0|\!)\ x := y + 1\ (\!|x > 0|\!)$
- $(\!|x \geq y|\!)\ x := x - y\ (\!|x \geq 0|\!)$
- $(\!|x \geq y|\!)\ x := x - y;\ y := -x\ (\!|y \leq 0|\!)$
- Swap without temp:
  $(\!|(x = x_0) \wedge (y = y_0)|\!)\ x := y - x;\ y := y - x;\ x := x + y\ (\!|(x := y_0) \wedge (y = x_0)|\!)$
- $(\!|\top|\!)\ \texttt{if}\ x < 2\ \texttt{then}\ x := 2\ \texttt{else}\ x := x\ (\!|x \geq 2|\!)$

$$\frac{}{(\!|G[E/x]|\!)\ x := E\ (\!|G|\!)}\ \textsc{Asg} \qquad \frac{(\!|F \wedge B|\!)\ C_1\ (\!|G|\!) \qquad (\!|F \wedge \neg B|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ \texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2\ (\!|G|\!)}\ \textsc{Cond}$$

$$\frac{(\!|F|\!)\ C_1\ (\!|\eta|\!) \qquad (\!|\eta|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ C_1; C_2\ (\!|G|\!)}\ \textsc{Comp} \qquad \frac{\vdash_{AR} F' \to F \qquad (\!|F|\!)\ C\ (\!|G|\!) \qquad \vdash_{AR} G \to G'}{(\!|F'|\!)\ C\ (\!|G'|\!)}\ \textsc{Impl}$$

Because of the rule for sequencing commands:

$$\frac{(\!|F|\!)\ C_1\ (\!|\eta|\!) \qquad (\!|\eta|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ C_1; C_2\ (\!|G|\!)}\ \textsc{Comp}$$

we often have to “invent” precodition η such that:

$$(\!|\eta???|\!)\ C_2\ (\!|F|\!)$$

We need to find the weakest precondition η.

Definition

η_1 weaker than η_2 whenever $\vdash_{AR} \eta_2 \rightarrow \eta_1$.

- it is harder to prove $(\!|\eta_1|\!)\ C_2\ (\!|G|\!)$ than it is to prove $(\!|\eta_2|\!)\ C_2\ (\!|G|\!)$
  - we assume a weaker property for the starting state of $\mathcal{C}_2$
  - **Therorem:** if $(\!|\eta_1|\!)\ C_2\ (\!|G|\!)$ then it follows $(\!|\eta_2|\!)\ C_2\ (\!|G|\!)$ (by Impl)
- but it is easier to prove $(\!|F|\!)\ C_1\ (\!|\eta_1|\!)$ than $(\!|F|\!)\ C_1\ (\!|\eta_2|\!)$
  - we need to prove a weaker property for the ending state of $\mathcal{C}_1$

Proof technique: start from the post condition of the program and work backwards through the code. Find the weakest precondition of each of the commands.

$$\frac{}{(\!|G[E/x]|\!)\ x := E\ (\!|G|\!)}\ \textsc{Asg} \qquad \frac{(\!|F \wedge B|\!)\ C_1\ (\!|G|\!) \qquad (\!|F \wedge \neg B|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ \texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2\ (\!|G|\!)}\ \textsc{Cond}$$

$$\frac{(\!|F|\!)\ C_1\ (\!|\eta|\!) \qquad (\!|\eta|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ C_1; C_2\ (\!|G|\!)}\ \textsc{Comp} \qquad \frac{\vdash_{AR} F' \to F \qquad (\!|F|\!)\ C\ (\!|G|\!) \qquad \vdash_{AR} G \to G'}{(\!|F'|\!)\ C\ (\!|G'|\!)}\ \textsc{Impl}$$

Proof technique: start from the post condition of the program and work backwards through the code. Find the weakest precondition of each of the commands.

- **Assignment:** $(\!|\eta?|\!)\ x := E\ (\!|G|\!)$
  set $\eta = G[E/x]$

$$\frac{}{(\!|G[E/x]|\!)\ x := E\ (\!|G|\!)}\ \textsc{Asg} \qquad \frac{(\!|F \wedge B|\!)\ C_1\ (\!|G|\!) \qquad (\!|F \wedge \neg B|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ \texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2\ (\!|G|\!)}\ \textsc{Cond}$$

$$\frac{(\!|F|\!)\ C_1\ (\!|\eta|\!) \qquad (\!|\eta|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ C_1; C_2\ (\!|G|\!)}\ \textsc{Comp} \qquad \frac{\vdash_{AR} F' \to F \qquad (\!|F|\!)\ C\ (\!|G|\!) \qquad \vdash_{AR} G \to G'}{(\!|F'|\!)\ C\ (\!|G'|\!)}\ \textsc{Impl}$$

Proof technique: start from the post condition of the program and work backwards through the code. Find the weakest precondition of each of the commands.

- **Assignment:** $(\!|\eta?|\!)\ x := E\ (\!|G|\!)$
  set $\eta = G[E/x]$
- **Composition:** $(\!|\eta?|\!)\ C_1; C_2\ (\!|G|\!)$
  find weakest precondition $(\!|\eta_1?|\!)\ C_2\ (\!|G|\!)$
  then find weakest precondition $(\!|\eta?|\!)\ C_1\ (\!|\eta_1|\!)$

$$\frac{}{(\!|G[E/x]|\!)\ x := E\ (\!|G|\!)}\ \textsc{Asg} \qquad \frac{(\!|F \wedge B|\!)\ C_1\ (\!|G|\!) \qquad (\!|F \wedge \neg B|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ \texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2\ (\!|G|\!)}\ \textsc{Cond}$$

$$\frac{(\!|F|\!)\ C_1\ (\!|\eta|\!) \qquad (\!|\eta|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ C_1; C_2\ (\!|G|\!)}\ \textsc{Comp} \qquad \frac{\vdash_{AR} F' \to F \qquad (\!|F|\!)\ C\ (\!|G|\!) \qquad \vdash_{AR} G \to G'}{(\!|F'|\!)\ C\ (\!|G'|\!)}\ \textsc{Impl}$$

Proof technique: start from the post condition of the program and work backwards through the code. Find the weakest precondition of each of the commands.

- **Assignment:** $(\!|\eta?|\!)\ x := E\ (\!|G|\!)$
  set $\eta = G[E/x]$
- **Composition:** $(\!|\eta?|\!)\ C_1; C_2\ (\!|G|\!)$
  find weakest precondition $(\!|\eta_1?|\!)\ C_2\ (\!|G|\!)$
  then find weakest precondition $(\!|\eta?|\!)\ C_1\ (\!|\eta_1|\!)$
- **Conditional:** $(\!|\eta?|\!)$ `if` B `then` C_1 `else` C_2 $(\!|G|\!)$
  find weakest precondition $(\!|\eta_1?|\!)\ C_1\ (\!|G|\!)$
  find weakest precondition $(\!|\eta_2?|\!)\ C_2\ (\!|G|\!)$
  set $\eta = (B \to \eta_1) \wedge (\neg B \to \eta_2)$

$$\frac{}{(\!|G[E/x]|\!)\ x := E\ (\!|G|\!)}\ \textsc{Asg} \qquad \frac{(\!|F \wedge B|\!)\ C_1\ (\!|G|\!) \qquad (\!|F \wedge \neg B|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ \texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2\ (\!|G|\!)}\ \textsc{Cond}$$

$$\frac{(\!|F|\!)\ C_1\ (\!|\eta|\!) \qquad (\!|\eta|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ C_1; C_2\ (\!|G|\!)}\ \textsc{Comp} \qquad \frac{\vdash_{AR} F' \to F \qquad (\!|F|\!)\ C\ (\!|G|\!) \qquad \vdash_{AR} G \to G'}{(\!|F'|\!)\ C\ (\!|G'|\!)}\ \textsc{Impl}$$

Proof technique: start from the post condition of the program and work backwards through the code. Find the weakest precondition of each of the commands.

For some program C, to prove

$$(\!| F |\!)\ C\ (\!| G |\!)$$

→ find the weakest precondition η from C and G

$$(\!| \eta |\!)\ C\ (\!| G |\!)$$

→ prove

$$\vdash_{\mathrm{AR}} F \rightarrow \eta$$

Prove the following Hoare triples:

- $(\!| y > 0 |\!)\ x := y + 1\ (\!| x > 0 |\!)$
- $(\!| x \geq y |\!)\ x := x - y\ (\!| x \geq 0 |\!)$
- $(\!| x \geq y |\!)\ x := x - y;\ y := -x\ (\!| y \leq 0 |\!)$
- Swap without temp:
  $(\!| (x = x_0) \wedge (y = y_0) |\!)\ x := y - x;\ y := y - x;\ x := x + y\ (\!| (x := y_0) \wedge (y = x_0) |\!)$
- $(\!| \top |\!)\ \texttt{if}\ x < 2\ \texttt{then}\ x := 2\ \texttt{else}\ x := x\ (\!| x \geq 2 |\!)$

$$\frac{}{(\!| G[E/x] |\!)\ x := E\ (\!| G |\!)}\ \textsc{Asg} \qquad \frac{(\!| F \wedge B |\!)\ C_1\ (\!| G |\!) \qquad (\!| F \wedge \neg B |\!)\ C_2\ (\!| G |\!)}{(\!| F |\!)\ \texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2\ (\!| G |\!)}\ \textsc{Cond}$$

$$\frac{(\!| F |\!)\ C_1\ (\!| \eta |\!) \qquad (\!| \eta |\!)\ C_2\ (\!| G |\!)}{(\!| F |\!)\ C_1; C_2\ (\!| G |\!)}\ \textsc{Comp} \qquad \frac{\vdash_{AR} F' \to F \qquad (\!| F |\!)\ C\ (\!| G |\!) \qquad \vdash_{AR} G \to G'}{(\!| F' |\!)\ C\ (\!| G' |\!)}\ \textsc{Impl}$$

Loops

$$\frac{(\!|G \wedge B|\!)\ C\ (\!|G|\!)}{(\!|G|\!)\ \texttt{while}\ B\ \{C\}\ (\!|G \wedge \neg B|\!)}\ \textsc{While}$$

We have to solve a recursive equation: we need a G that holds before and after C, and therefore before and after the entire while loop.

G is the invariant of the loop.

→ G holds before and after every iteration of the loop.

G usually has to be **imagined** (requires intuition).

Prove that $\vdash_{\text{par}} (\!|x > 0|\!)$ Fact1 $(\!|y = x!|\!)$ when Fact1 is the program:

```
y := 1;
z := 0;
while (z != x) {
  z := z + 1;
  y := y * z;
}
```

$$\frac{}{(\!|G[E/x]|\!)\ x := E\ (\!|G|\!)}\ \textsc{Asg} \qquad \frac{(\!|F \wedge B|\!)\ C_1\ (\!|G|\!) \qquad (\!|F \wedge \neg B|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ \texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2\ (\!|G|\!)}\ \textsc{Cond}$$

$$\frac{(\!|F|\!)\ C_1\ (\!|\eta|\!) \qquad (\!|\eta|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ C_1; C_2\ (\!|G|\!)}\ \textsc{Comp} \qquad \frac{\vdash_{\text{AR}} F' \to F \qquad (\!|F|\!)\ C\ (\!|G|\!) \qquad \vdash_{\text{AR}} G \to G'}{(\!|F'|\!)\ C\ (\!|G'|\!)}\ \textsc{Impl}$$

$$\frac{(\!|G \wedge B|\!)\ C\ (\!|G|\!)}{(\!|G|\!)\ \texttt{while}\ B\ \{C\}\ (\!|G \wedge \neg B|\!)}\ \textsc{While}$$

$(\!\| x > 0 \|\!)$	
$(\!\| \top \|\!)$	implied
$(\!\| 1 = 0! \|\!)$	implied
`y := 1;`	
$(\!\| y = 0! \|\!)$	ASG
`z := 0;`	
$(\!\| y = z! \|\!)$	ASG
`while(z != x){`	
$(\!\| (y = z!) \wedge (z \neq x) \|\!)$	WHILE
$(\!\| y * (z + 1) = z! * (z + 1) \|\!)$	implied
$(\!\| y * (z + 1) = (z + 1)! \|\!)$	implied
`z := z + 1;`	
$(\!\| y * z = z! \|\!)$	ASG
`y := y * z;`	
$(\!\| y = z! \|\!)$	ASG
`}`	
$(\!\| (y = z!) \wedge \neg(x \neq z) \|\!)$	WHILE
$(\!\| y = x! \|\!)$	implied

Prove that $\vdash_{par}$ $(\!|(x = x_0) \wedge (x \geq 0)|\!)$ Fact2 $(\!|y = x_0!|\!)$ when Fact2 is the program:

```
y := 1;
while (x != 0) {
  y := y * x;
  x := x - 1;
}
```

$$\frac{}{(\!|G[E/x]|\!)\ x := E\ (\!|G|\!)}\ \textsc{Asg} \qquad \frac{(\!|F \wedge B|\!)\ C_1\ (\!|G|\!) \qquad (\!|F \wedge \neg B|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ \texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2\ (\!|G|\!)}\ \textsc{Cond}$$

$$\frac{(\!|F|\!)\ C_1\ (\!|\eta|\!) \qquad (\!|\eta|\!)\ C_2\ (\!|G|\!)}{(\!|F|\!)\ C_1; C_2\ (\!|G|\!)}\ \textsc{Comp} \qquad \frac{\vdash_{AR} F' \to F \qquad (\!|F|\!)\ C\ (\!|G|\!) \qquad \vdash_{AR} G \to G'}{(\!|F'|\!)\ C\ (\!|G'|\!)}\ \textsc{Impl}$$

$$\frac{(\!|G \wedge B|\!)\ C\ (\!|G|\!)}{(\!|G|\!)\ \texttt{while}\ B\ \{C\}\ (\!|G \wedge \neg B|\!)}\ \textsc{While}$$