

# Program Verification

## While Loops

Alice Gao  
Lecture 20

Based on work by J. Buss, L. Kari, A. Lubiw, B. Bonakdarpour, D. Maftuleac, C. Roberts, R. Treffer, and P. Van Beek

# Outline

## Program Verification: While Loops

- Learning Goals

- Proving Partial Correctness - Example 1

- Proving Partial Correctness - Example 2

- Proving Termination

- Revisiting the Learning Goals

# Learning Goals

By the end of this lecture, you should be able to:

Partial correctness for while loops

- ▶ Determine whether a given formula is an invariant for a while loop.
- ▶ Find an invariant for a given while loop.
- ▶ Prove that a Hoare triple is satisfied under partial correctness for a program containing while loops.

Total correctness for while loops

- ▶ Determine whether a given formula is a variant for a while loop.
- ▶ Find a variant for a given while loop.
- ▶ Prove that a Hoare triple is satisfied under total correctness for a program containing while loops.

# Proving Total Correctness of While Loops

- ▶ Partial correctness
- ▶ Termination

## Proving Partial Correctness of While Loops

$\{P\}$   
 $\{I\}$  implied (A)  
**while** ( B ) {  
     $\{(I \wedge B)\}$  partial-**while**  
    C  
     $\{I\}$  <justify based on C – a subproof>  
}  
 $\{(I \wedge (\neg B))\}$  partial-**while**  
 $\{Q\}$  implied (B)

Proof of implied (A):  $(P \rightarrow I)$

Proof of implied (B):  $((I \wedge (\neg B)) \rightarrow Q)$

$I$  is called a **loop invariant**. We need to determine  $I$ !

# What is a loop invariant?

A loop invariant is:

- ▶ A relationship among the variables. (A predicate formula involving the variables.)
- ▶ The word “invariant” means something that does not change.
- ▶ It is true before the loop begins.
- ▶ It is true at the start of every iteration of the loop and at the end of every iteration of the loop.
- ▶ It is true after the loop ends.

## Proving partial correctness of while loops

Indicate the places in the program where the loop invariant is true.

```
 $\langle (x \geq 0) \rangle$ 
```

```
y = 1;
```

```
z = 0;
```

```
while (z != x) {
```

```
    z = z + 1;
```

```
    y = y * z;
```

```
}
```

```
 $\langle (y = x!) \rangle$ 
```

# Proving partial correctness of a while loop

Steps to follow:

- ▶ Find a loop invariant.
- ▶ Complete the annotations.
- ▶ Prove any implied's.

How do we find a loop invariant???

# How do we find a loop invariant?

First, we need to understand the purpose of an invariant.

- ▶ The postcondition is the ultimate goal of our while loop.
- ▶ At every iteration, we are making progress towards the postcondition.
- ▶ The invariant is describing the progress we are making at every iteration.

## Partial While - Example 1

### Example 1:

Prove that the following triple is satisfied under partial correctness.

$\langle\langle x \geq 0 \rangle\rangle$

$y = 1;$

$z = 0;$

**while**  $(z \neq x)$  {

$z = z + 1;$

$y = y * z;$

}

$\langle\langle y = x! \rangle\rangle$

## Finding a loop invariant

Step 1: Write down the values of all the variables every time the while test is reached.

```
((x ≥ 0))
```

```
y = 1;
```

```
z = 0;
```

```
while (z != x) {
```

```
    z = z + 1;
```

```
    y = y * z;
```

```
}
```

```
((y = x!))
```

## Finding a loop invariant

Step 2: Find relationships among the variables that are true for every while test. These are our candidate invariants.

Come up with some invariants in the next 2 minutes.

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

x	z	y
5	0	1 = 0!
5	1	1 = 1!
5	2	2 = 2!
5	3	6 = 3!
5	4	24 = 4!
5	5	120 = 5!

## CQ 1 Is this a loop invariant?

**CQ 1:** Is  $(\neg(z = x))$  a loop invariant?

(A) Yes (B) No (C) I don't know...

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

x	z	y
5	0	1 = 0!
5	1	1 = 1!
5	2	2 = 2!
5	3	6 = 3!
5	4	24 = 4!
5	5	120 = 5!

## CQ 2 Is this a loop invariant?

**CQ 2:** Is  $(z \leq x)$  a loop invariant?

(A) Yes (B) No (C) I don't know...

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

x	z	y
5	0	1 = 0!
5	1	1 = 1!
5	2	2 = 2!
5	3	6 = 3!
5	4	24 = 4!
5	5	120 = 5!

## CQ 3 Is this a loop invariant?

**CQ 3:** Is  $(y = z!)$  a loop invariant?

(A) Yes (B) No (C) I don't know...

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

x	z	y
5	0	1 = 0!
5	1	1 = 1!
5	2	2 = 2!
5	3	6 = 3!
5	4	24 = 4!
5	5	120 = 5!

## CQ 4 Is this a loop invariant?

**CQ 4:** Is  $(y = x!)$  a loop invariant?

(A) Yes (B) No (C) I don't know...

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

x	z	y
5	0	1 = 0!
5	1	1 = 1!
5	2	2 = 2!
5	3	6 = 3!
5	4	24 = 4!
5	5	120 = 5!

## CQ 5 Is this a loop invariant?

**CQ 5:** Is  $((z \leq x) \wedge (y = z!))$  a loop invariant?

(A) Yes (B) No (C) I don't know...

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

x	z	y
5	0	1 = 0!
5	1	1 = 1!
5	2	2 = 2!
5	3	6 = 3!
5	4	24 = 4!
5	5	120 = 5!

## Finding a loop invariant

Step 3: Try each candidate invariant until we find one that works for our proof.

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

x	z	y
5	0	1 = 0!
5	1	1 = 1!
5	2	2 = 2!
5	3	6 = 3!
5	4	24 = 4!
5	5	120 = 5!

# How do we find an invariant?

A recap of the steps to find an invariant:

- ▶ Write down the values of all the variables every time the while test is reached.
- ▶ Find relationships among the variables that are true for every while test. These are our candidate invariants.
- ▶ Try each candidate invariant until we find one that works for our proof.

## CQ 6 Which loop invariant would you like to try first?

**CQ 6:** Which loop invariant would you like to try first?

(A)  $(z \leq x)$

(B)  $(y = z!)$

(C)  $((z \leq x) \wedge (y = z!))$

## Partial While - Example 1 ( $(z \leq x)$ as the invariant)

$\langle\langle x \geq 0 \rangle\rangle$	
$\langle\langle 0 \leq x \rangle\rangle$	implied (A)
$y = 1;$	
$\langle\langle 0 \leq x \rangle\rangle$	assignment
$z = 0;$	
$\langle\langle z \leq x \rangle\rangle$	assignment
<b>while</b> ( $z \neq x$ ) {	
$\langle\langle (z \leq x) \wedge (\neg(z = x)) \rangle\rangle$	partial- <b>while</b>
$\langle\langle z + 1 \leq x \rangle\rangle$	implied (B)
$z = z + 1;$	
$\langle\langle z \leq x \rangle\rangle$	assignment
$y = y * z;$	
$\langle\langle z \leq x \rangle\rangle$	assignment
}	
$\langle\langle (z \leq x) \wedge (\neg(\neg(z = x))) \rangle\rangle$	partial- <b>while</b>
$\langle\langle y = x! \rangle\rangle$	implied (C)

## CQ 7 Is there a proof for implied (A)?

We used  $(z \leq x)$  as the invariant.

**CQ 7:** Is there a proof for implied (A)?

$$((x \geq 0) \rightarrow (0 \leq x))$$

- (A) Yes
- (B) No
- (C) I don't know.

## CQ 8 Is there a proof for implied (B)?

We used  $(z \leq x)$  as the invariant.

**CQ 8:** Is there a proof for implied (B)?

$$(((z \leq x) \wedge (\neg(z = x)))) \rightarrow (z + 1 \leq x))$$

- (A) Yes
- (B) No
- (C) I don't know.

## CQ 9 Is there a proof for implied (C)?

We used  $(z \leq x)$  as the invariant.

**CQ 9:** Is there a proof for implied (C)?

$$(((z \leq x) \wedge (\neg(\neg(z = x)))) \rightarrow (y = x!))$$

- (A) Yes
- (B) No
- (C) I don't know.

## Example 1: Summary of invariants

Which invariant leads to a valid proof?

- ▶  $(z \leq x)$  does NOT lead to a valid proof.
- ▶  $(y = z!)$  does lead to a valid proof.
- ▶  $((z \leq x) \wedge (y = z!))$  does lead to a valid proof.

## Partial While - Example 2

### Example 2:

Prove that the following triple is satisfied under partial correctness.

```
 $\langle\langle x \geq 0 \rangle\rangle$   
y = 1;  
z = 0;  
while (z < x) {  
    z = z + 1;  
    y = y * z;  
}  
 $\langle\langle y = x! \rangle\rangle$ 
```

Let's try using  $(y = z!)$  as the invariant in our proof.

## CQ 10 Which invariant leads to a valid proof?

**CQ 10:** For example 2, which invariant leads to a valid proof?

(A)  $(z \leq x)$

(B)  $(y = z!)$

(C)  $((z \leq x) \wedge (y = z!))$

(D) Two of (A), (B), and (C).

(E) All of (A), (B), and (C).

## Which invariant leads to a valid proof?

To check whether an invariant leads to a valid proof,  
we need to check whether all of the implied's can be proved.

## CQ 11 Is there a proof for implied (A)?

We used  $(y = z!)$  as the invariant.

**CQ 11:** Is there a proof for implied (A)?

$$((x \geq 0) \rightarrow (1 = 0!))$$

- (A) Yes
- (B) No
- (C) I don't know.

## CQ 12 Is there a proof for implied (B)?

We used  $(y = z!)$  as the invariant.

**CQ 12:** Is there a proof for implied (B)?

$$(((y = z!) \wedge (z < x)) \rightarrow (y * (z + 1) = (z + 1)!))$$

- (A) Yes
- (B) No
- (C) I don't know.

## CQ 13 Is there a proof for implied (C)?

We used  $(y = z!)$  as the invariant.

**CQ 13:** Is there a proof for implied (C)?

$$(((y = z!) \wedge (\neg(z < x))) \rightarrow (y = x!))$$

- (A) Yes
- (B) No
- (C) I don't know.

## CQ 14 Is there a proof for implied (C)?

We used  $((y = z!) \wedge (z \leq x))$  as the invariant.

**CQ 14:** Is there a proof for implied (C)?

$$(((y = z!) \wedge (z \leq x)) \wedge (\neg(z < x))) \rightarrow (y = x!)$$

- (A) Yes
- (B) No
- (C) I don't know.

## CQ 15 Which invariant leads to a valid proof?

**CQ 15:** For example 2, which invariant leads to a valid proof?

(A)  $(z \leq x)$

(B)  $(y = z!)$

(C)  $((z \leq x) \wedge (y = z!))$

(D) Two of (A), (B), and (C).

(E) All of (A), (B), and (C).

## Example 2: Summary of invariants

Which invariant leads to a valid proof?

- ▶  $(z \leq x)$  does NOT lead to a valid proof.
- ▶  $(y = z!)$  does NOT lead to a valid proof.
- ▶  $((z \leq x) \wedge (y = z!))$  does lead to a valid proof.

# Proving Termination

Find an integer expression that

- ▶ is non-negative before the loop starts, at every iteration of the loop, and after the loop ends.
- ▶ decreases by at least 1 at every iteration of the loop.

This integer expression is called a **variant** (something that changes).

The loop must terminate because a non-negative integer can decrease by 1 a finite number of times.

## Example 2: Finding a variant

### Example 2:

Prove that the following program terminates.

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

How do we find a variant? The loop guard ( $z < x$ ) helps.

## Example 2: Proof of Termination

Consider the variant  $(x - z)$ .

Before the loop starts,  $(x - z) \geq 0$  because the precondition is  $(x \geq 0)$  and the second assignment mutates  $z$  to be 0.

During every iteration of the loop,  $(x - z)$  decreases by 1 because  $x$  does not change and  $z$  increases by 1.

Thus,  $x - z$  will eventually reach 0.

When  $x - z = 0$ , the loop guard  $z < x$  will terminate the loop.

# Revisiting the learning goals

By the end of this lecture, you should be able to:

Partial correctness for while loops

- ▶ Determine whether a given formula is an invariant for a while loop.
- ▶ Find an invariant for a given while loop.
- ▶ Prove that a Hoare triple is satisfied under partial correctness for a program containing while loops.

Total correctness for while loops

- ▶ Determine whether a given formula is a variant for a while loop.
- ▶ Find a variant for a given while loop.
- ▶ Prove that a Hoare triple is satisfied under total correctness for a program containing while loops.