Proofs of Correctness: An Introduction to Axiomatic Verification

CEN 5035
Software Engineering

Prepared by
Stephen M. Thebaut, Ph.D.
University of Florida
Outline

• Introduction
• Weak correctness predicate
• Assignment statements
• Sequencing
• Selection statements
• Iteration
Introduction

• *What is Axiomatic Verification?*

A formal method of reasoning about the *functional* correctness of a structured, sequential program by tracing its state changes from an initial (i.e., pre-) condition to a final (i.e., post-) condition according to a set of self-evident rules (i.e., axioms).
Introduction (cont’d)

• *What is its primary goal?*
  
  To provide a means for “proving” (or “disproving”) the functional correctness of a sequential program with respect to its (formal) specification.
Introduction (cont’d)

• What are the benefits of studying axiomatic verification?
  – Understanding its *limitations*.
  – Deeper *insights* into programming and program structures.
  – Criteria for judging both programs and programming languages.
  – The ability to formally verify small (or parts of large) sequential programs.
Introduction (cont’d)

- Bottom line: even if you never attempt to “prove” a program correct outside this course, the study of formal verification should change the way you write and read programs.
Weak Correctness Predicate

- To prove that program S is (weakly) correct with respect to pre-condition P and post-condition Q, it is sufficient to show: \{P\} S \{Q\}.

- Interpretation of \{P\} S \{Q\}: “if the input (initial state) satisfies pre-condition P and (if) program S executes and terminates, then the output (final state) must satisfy post-condition Q.”
Weak Correctness Predicate (cont’d)

• Note that \{P\} S \{Q\} is really just a “double conditional” of the form:

\[(A \land B) \implies C\]

where A is “P holds before executing S”, B is “S terminates”, and C is “Q holds after executing S”.

• Therefore, what is the one and only case (in terms of the values of A, B, and C) for which \{P\} S \{Q\} is false?
Weak Correctness Predicate (cont’d)

• Thus, \( \{P\} S \{Q\} \) is true unless \( Q \) could be false if \( S \) terminates, given that \( P \) held before \( S \) executes.

• What are the truth values of the following assertions?

\[
(1) \ {x=1} y := x+1 \ {y>0}
\]
Weak Correctness Predicate (cont’d)

• Thus, \( \{P\} S \{Q\} \) is true unless \( Q \) could be false if \( S \) terminates, given that \( P \) held before \( S \) executes.

• What are the truth values of the following assertions?

\[
(1) \quad \{x=1\} y := x+1 \{y>0\}
\]

True, because if \( P \) holds initially, \( Q \) must hold when \( S \) terminates.
Thus, \{P\} S \{Q\} is true unless Q could be false if S terminates, given that P held before S executes.

What are the truth values of the following assertions?

(2) \{x>0\} x := x-1 \{x>0\}
Weak Correctness Predicate (cont’d)

• Thus, \{P\} S \{Q\} is true unless Q could be false if S terminates, given that P held before S executes.

• What are the truth values of the following assertions?

(2) \{x>0\} x := x-1 \{x>0\}

False, because Q may not hold when S terminates given that P holds initially.
Weak Correctness Predicate (cont’d)

- Thus, \( \{P\} \ S \ \{Q\} \) is \textit{true} unless \( Q \) \textit{could be false} if \( S \) terminates, given that \( P \) held before \( S \) executes.
- What are the truth values of the following assertions?

\[
(3) \ \{1=2\} \ k := 5 \ \{k<0\}
\]
Weak Correctness Predicate (cont’d)

• Thus, \{P\} S \{Q\} is *true* unless Q *could be false* if S terminates, given that P held before S executes.

• What are the truth values of the following assertions?

\[(3) \{1=2\} k := 5 \{k<0\}\]

*True (vacuously), since P cannot hold* before S executes.
Weak Correctness Predicate (cont’d)

• Thus, \{P\} S \{Q\} is *true* unless Q *could be false* if S terminates, given that P held before S executes.

• What are the truth values of the following assertions?

  (4) \{true\} while x <> 5 do x := x-1 \{x=5\}

  (Hint: When will S terminate?)
Weak Correctness Predicate (cont’d)

• Thus, \( \{P\} S \{Q\} \) is true unless \( Q \) could be false if \( S \) terminates, given that \( P \) held before \( S \) executes.

• What are the truth values of the following assertions?
  
  (4) \( \{true\} \) while \( x <> 5 \) do \( x := x-1 \) \( \{x=5\} \)  
  
  (Hint: When will \( S \) terminate?)

  \( True \), since \( Q \) must hold if \( S \) terminates.
Weak Correctness Predicate (cont’d)

• We now consider techniques for proving that such assertions hold for structured programs comprised of assignment statements, if-then (-else) statements, and while loops.

(Why these particular constructs?)
Reasoning about Assignment Statements

• For each of the following pre-conditions, P, and assignment statements, S, identify a “strong” post-condition, Q, such that \( \{P\} S \{Q\} \) would hold.

• A “strong” post-condition captures all after-execution state information of interest.

• We won’t bother with propositions such as \( X=X' \) (“the final value of X is the same as the initial value of X”) for the time being.
Reasoning about Assignment Statements (cont’d)

\[
\begin{align*}
\{P\} & \quad S & \quad \{Q\} \\
\{J=6\} & \quad K := 3 & \\
\{J=6\} & \quad J := J+2 & \\
\{A<B\} & \quad \text{Min} := A & \\
\{X<0\} & \quad Y := -X & 
\end{align*}
\]
Reasoning about Assignment Statements (cont’d)

\[
\begin{align*}
\{P\} \quad S \\
\{J=6\} \quad K := 3 \\
\{J=6\} \quad J := J+2 \\
\{A<B\} \quad \text{Min} := A \\
\{X<0\} \quad Y := -X \\
\{Q\} \\
\{J=6 \land K=3\}
\end{align*}
\]
Reasoning about Assignment Statements (cont’d)

\[
\begin{array}{|c|c|c|}
\hline
\{P\} & S & \{Q\} \\
\{J=6\} & K := 3 & \{J=6 \land K=3\} \\
\{J=6\} & J := J+2 & \\
\{A<B\} & \text{Min} := A & \\
\{X<0\} & Y := -X & \\
\hline
\end{array}
\]
Reasoning about Assignment Statements (cont’d)

\[
\begin{align*}
\{ \text{P} \} & \quad \{ \text{Q} \} \\
\{ J=6 \} & \quad \{ J=6 \land K=3 \} \\
\{ A<B \} & \\
\{ X<0 \} & \\
\end{align*}
\]

\[
\begin{align*}
S & \quad \text{J := J+2} \\
\text{K := 3} & \\
\text{Min := A} & \\
\text{Y := -X} & \\
\end{align*}
\]
Reasoning about Assignment Statements (cont’d)

\[
\begin{array}{c|c|c}
\{P\} & S & \{Q\} \\
\{J=6\} & K := 3 & \{J=6 \land K=3\} \\
\{J=6\} & J := J+2 & \{J=8\} \\
\{A<B\} & \text{Min} := A & \\
\{X<0\} & Y := -X & \\
\end{array}
\]
Reasoning about Assignment Statements (cont’d)

\[
\begin{align*}
\{P\} & \quad \{Q\} \\
\{J=6\} & \quad \{J=6 \land K=3\} \\
\{J=6\} & \quad \{J=8\} \\
\{A<B\} & \quad \{A<B \land Min=A\} \\
\{X<0\} & \quad \{A<B \land Min=A\} \\
S & \quad K := 3 \\
J := J+2 & \\
Min := A & \\
Y := -X & 
\end{align*}
\]
## Reasoning about Assignment Statements (cont’d)

<table>
<thead>
<tr>
<th>${P}$</th>
<th>$S$</th>
<th>${Q}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${J=6}$</td>
<td>$K := 3$</td>
<td>${J=6 \land K=3}$</td>
</tr>
<tr>
<td>${J=6}$</td>
<td>$J := J+2$</td>
<td>${J=8}$</td>
</tr>
<tr>
<td>${A&lt;B}$</td>
<td>$\text{Min} := A$</td>
<td>${A&lt;B \land \text{Min}=A}$</td>
</tr>
<tr>
<td>${X&lt;0}$</td>
<td>$Y := -X$</td>
<td></td>
</tr>
</tbody>
</table>
Reasoning about Assignment Statements (cont’d)

\[
\begin{align*}
\{P\} & \quad S & \quad \{Q\} \\
\{J=6\} & \quad K := 3 & \{J=6 \land K=3\} \\
\{J=6\} & \quad J := J+2 & \{J=8\} \\
\{A<B\} & \quad \text{Min} := A & \{A<B \land \text{Min}=A\} \\
\{X<0\} & \quad Y := -X & \{X<0 \land Y=-X\}
\end{align*}
\]
Reasoning about Assignment Statements (cont’d)

• For each of the following post-conditions, Q, and assignment statements, S, identify a “weak” pre-condition, P, such that \{P\} S \{Q\} would hold.

(A “weak” pre-condition reflects only what needs to be true beforehand.)
Reasoning about Assignment Statements (cont’d)

\[
\begin{align*}
\{P\} & \quad S & \quad \{Q\} \\
I := 4 & & \{J=7 \land I=4\} \\
I := 4 & & \{I=4\} \\
I := 4 & & \{I=17\} \\
Y := X+3 & & \{Y=10\}
\end{align*}
\]
Reasoning about Assignment Statements (cont’d)

<table>
<thead>
<tr>
<th>{P}</th>
<th>{Q}</th>
</tr>
</thead>
<tbody>
<tr>
<td>{J=7}</td>
<td>{J=7 \land I=4}</td>
</tr>
<tr>
<td>I := 4</td>
<td>I := 4</td>
</tr>
<tr>
<td>I := 4</td>
<td>I := 4</td>
</tr>
<tr>
<td>I := 4</td>
<td>I := 17</td>
</tr>
<tr>
<td>Y := X + 3</td>
<td>Y := 10</td>
</tr>
</tbody>
</table>
Reasoning about Assignment Statements (cont’d)

<table>
<thead>
<tr>
<th><strong>P</strong></th>
<th><strong>S</strong></th>
<th><strong>Q</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>(J = 7)</td>
<td>(I := 4)</td>
<td>(J = 7 \land I = 4)</td>
</tr>
<tr>
<td>(I := 4)</td>
<td>(I = 4)</td>
<td>(I = 4)</td>
</tr>
<tr>
<td>(I := 4)</td>
<td>(I = 17)</td>
<td>(Y = 10)</td>
</tr>
<tr>
<td>(Y := X + 3)</td>
<td>()</td>
<td>()</td>
</tr>
</tbody>
</table>
Reasoning about Assignment Statements (cont’d)

<table>
<thead>
<tr>
<th>${P}$</th>
<th>$S$</th>
<th>${Q}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${J=7}$</td>
<td>$I := 4$</td>
<td>${J=7 \land I=4}$</td>
</tr>
<tr>
<td>${true}$</td>
<td>$I := 4$</td>
<td>${I=4}$</td>
</tr>
<tr>
<td></td>
<td>$I := 4$</td>
<td>${I=17}$</td>
</tr>
<tr>
<td></td>
<td>$Y := X+3$</td>
<td>${Y=10}$</td>
</tr>
</tbody>
</table>
Reasoning about Assignment Statements (cont’d)

<table>
<thead>
<tr>
<th>${P}$</th>
<th>${Q}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${J=7}$</td>
<td>${J=7 \land I=4}$</td>
</tr>
<tr>
<td>${\text{true}}$</td>
<td>${I=4}$</td>
</tr>
<tr>
<td>$I := 4$</td>
<td>$I := 4$</td>
</tr>
<tr>
<td>$I := 4$</td>
<td>$I := 4$</td>
</tr>
<tr>
<td>$Y := X+3$</td>
<td>$Y := 10$</td>
</tr>
</tbody>
</table>
### Reasoning about Assignment Statements (cont’d)

<table>
<thead>
<tr>
<th>${P}$</th>
<th>$S$</th>
<th>${Q}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${J=7}$</td>
<td>$I := 4$</td>
<td>${J=7 \land I=4}$</td>
</tr>
<tr>
<td>${true}$</td>
<td>$I := 4$</td>
<td>${I=4}$</td>
</tr>
<tr>
<td>${false}$</td>
<td>$I := 4$</td>
<td>${I=17}$</td>
</tr>
<tr>
<td>$Y := X+3$</td>
<td>${Y=10}$</td>
<td></td>
</tr>
</tbody>
</table>
Reasoning about Assignment Statements (cont’d)

\[
\begin{align*}
\{P\} & \quad S & \quad \{Q\} \\
\{J=7\} & \quad I := 4 & \quad \{J=7 \land I=4\} \\
\{true\} & \quad I := 4 & \quad \{I=4\} \\
\{false\} & \quad I := 4 & \quad \{I=17\} \\
Y := X+3 & \quad \{Y=10\}
\end{align*}
\]
Reasoning about Assignment Statements (cont’d)

{P} {J=7} {true} {false} {X=7}  
{Q} {J=7 ∧ I=4} {I=4} {I=17} {Y=10}
Reasoning about Sequencing

- In general: if you know \{P\} S_1 \{R\} and you know \{R\} S_2 \{Q\} then you know \{P\} S_1 ; S_2 \{Q\}.

  (So, to prove \{P\} S_1 ; S_2 \{Q\}, find \{R\}.)
Example 1

- Prove the assertion:

\{A=5\} \ B := A+2; \ C := B-A; \ D := A-C \ \{A=5 \land D=3\}
Example 1

• Prove the assertion:

\[ \{A=5\} \ B := A+2; \ C := B-A; \ D := A-C \ \{A=5 \land D=3\} \]

\[ \{A=5\} \ B := A+2; \]
Example 1

• Prove the assertion:

\{A=5\} B := A+2; C := B-A; D := A-C \ {A=5 & D=3}\n
\{A=5\}
B := A+2;
\{A=5 & B=7\}
Example 1

• Prove the assertion:

\[ \{A=5\} \ \text{B := A+2; C := B-A; D := A-C \ \{A=5 \land D=3\}} \]

\[ \{A=5\} \]
  \[
  \text{B := A+2;}
  \{A=5 \land B=7\}
  \]
  \[
  \text{C := B-A;}
  \]
Example 1

- Prove the assertion:

$$\{A=5\} \quad B := A+2; \quad C := B-A; \quad D := A-C \quad \{A=5 \land D=3\}$$

$$\{A=5\}$$
$$\quad B := A+2;$$
$$\{A=5 \land B=7\}$$
$$\quad C := B-A;$$
$$\{A=5 \land B=7 \land C=2\}$$
Example 1

- Prove the assertion:

\{A=5\} \quad B := A+2; \quad C := B-A; \quad D := A-C \quad \{A=5 \land D=3\}

\{A=5\}
   \quad B := A+2;
\{A=5 \land B=7\}
   \quad C := B-A;
\{A=5 \land B=7 \land C=2\}
   \quad D := A-C
Example 1

- Prove the assertion:

\{A=5\} B := A+2; C := B-A; D := A-C \{A=5 \land D=3\}

\begin{align*}
\{A=5\} \\
\quad B & := A+2; \\
\{A=5 \land B=7\} \\
\quad C & := B-A; \\
\{A=5 \land B=7 \land C=2\} \\
\quad D & := A-C \\
\{A=5 \land B=7 \land C=2 \land D=3\}
\end{align*}
Example 1

• Prove the assertion:

\{A=5\} B := A+2; C := B-A; D := A-C \quad \{A=5 \land D=3\}

\{A=5\}
  B := A+2;
\{A=5 \land B=7\}
  C := B-A;
\{A=5 \land B=7 \land C=2\}
  D := A-C
\{A=5 \land B=7 \land C=2 \land D=3\}
Example 1

• Prove the assertion:

\{A=5\} \quad \text{B := A+2; C := B-A; D := A-C} \quad \text{\{A=5 \land D=3\}}

\{A=5\}
\quad \text{B := A+2;}
\{A=5 \land B=7\}
\quad \text{C := B-A;}
\{A=5 \land B=7 \land C=2\}
\quad \text{D := A-C}
\{A=5 \land B=7 \land C=2 \land D=3\} \implies \{A=5 \land D=3\}
Reasoning about If_then_else Statements

- Consider the assertion:
  \{P\} if \(b\) then \(S_1\) else \(S_2\) \{Q\}

- What are the necessary conditions for this assertion to hold?
Necessary Conditions: If_then_else
Necessary Conditions: If\_then\_else

\[
\begin{align*}
\{P\} & \quad \text{if } b = \text{T} \\
\{P \land b\} & \quad \text{if } b = \text{F} \\
\{Q\} & \\
\end{align*}
\]
Necessary Conditions: If_then_else

\{P\} \rightarrow T \rightarrow b \rightarrow S_1 \rightarrow \{Q\}

\{P \land b\} S_1 \{Q\}

and
Necessary Conditions: If_then_else

\[
\begin{align*}
&P \land b \implies S_1 \implies Q \\
&P \land \neg b \implies S_2 \implies Q
\end{align*}
\]
Reasoning about If then Statements

- Consider the assertion:
  \( \{P\} \text{ if } b \text{ then } S \{Q\} \)

- What are the *necessary* conditions for this assertion to hold?
Necessary Conditions: If_then
Necessary Conditions: If_then

\( \{P\} \quad \{P \land b\} \quad \{Q\} \)
Necessary Conditions: If_then

\{P\} \downarrow T \rightarrow \{P \land b\} \downarrow \rightarrow \{Q\}

\{Q\} \downarrow F \rightarrow \{Q\}

and
Necessary Conditions: If_then

\[
\{P\} \rightarrow \{P \land b \} \land \{Q\} \\
\text{and} \\
(P \land \neg b) \Rightarrow Q
\]
Example 2

• Prove the assertion:

\[ \{Z = B\} \text{ if } A > B \text{ then } Z := A \{Z = \text{Max}(A, B)\} \]
Example 2

• Prove the assertion:

\[
\begin{align*}
P & \quad b & \quad S & \quad Q \\
\{Z=B\} & \text{if } A>B & \text{then } Z := A & \{Z=\text{Max}(A,B)\}
\end{align*}
\]
Example 2

• Prove the assertion:

\[ \{Z=B\} \text{ if } A>B \text{ then } Z := A \{Z=\text{Max}(A,B)\} \]

(1) \{P \land b \} S \{Q\}:

(2) \,(P \land \neg b) \Rightarrow Q:
Example 2

• Prove the assertion:

\[ Z = B \] if \( A > B \) then \( Z := A \) \( \{ Z = \text{Max}(A,B) \} \)

(1) \( \{ P \land b \} \implies \{ Q \} : \)

\( \{ Z = B \land A > B \} \implies \{ Z := A \} \)

(2) \( (P \land \neg b) \implies Q : \)
Example 2

- Prove the assertion:

\[
\begin{align*}
P & \quad b & S & \quad Q \\
\{Z=B\} \text{ if } A>B \text{ then } Z := A & \{Z=\text{Max}(A,B)\}
\end{align*}
\]

(1) \{P \land b \} \implies \{Q\}:

\[
\{Z=B \land A>B\} \quad Z := A & \{Z=A \land A>B\}
\]

(2) \(P \land \neg b\) \implies Q:
Example 2

- Prove the assertion:

\[
\begin{align*}
\{Z=B\} & \text{ if } A>B \text{ then } Z := A \quad \{Z=\text{Max}(A,B)\} \\
\end{align*}
\]

(1) \(\{P \land b\} \Rightarrow \{Q\}: \sqrt{\ }

\[
\begin{align*}
\{Z=B \land A>B\} & \quad Z := A \quad \{Z=A \land A>B\} \Rightarrow Q \\
\end{align*}
\]

(2) \(\{P \land \neg b\} \Rightarrow Q:\)
Example 2

• Prove the assertion:

\[\begin{align*}
\text{if } A > B \text{ then } Z := A \{Z = \text{Max}(A, B)\}
\end{align*}\]

(1) \(\{P \land b\} \implies Q\): \(\checkmark\)

\[\{Z = B \land A > B\} \implies (Z = A \land A > B) \implies Q\]

(2) \(\{P \land \neg b\} \implies Q\):

\[\{Z = B \land A \leq B\} \implies ?\]
Example 2

Prove the assertion:

\{Z=B\} \text{ if } A>B \text{ then } Z := A \{Z=\text{Max}(A,B)\}

(1) \{P \land b \} S \{Q\}: \checkmark

\{Z=B \land A>B\} Z := A \{Z=A \land A>B\} \Rightarrow Q

(2) (P \land \neg b) \Rightarrow Q: \checkmark

(Z=B \land A \leq B) \Rightarrow Q
Proof Rules

• Before proceeding to while loops, let’s capture our previous reasoning about sequencing and selection statements in appropriate rules of inference (ROI).

ROI for Sequencing:

\[
\begin{align*}
\{P\} S_1 \{R\}, \{R\} S_2 \{Q\} \\
\{P\} S_1; S_2 \{Q\}
\end{align*}
\]
Proof Rules (cont’d)

ROI for *if_then_else* statement:

\[
\{P \land b \} S_1 \{Q\}, \{P \land \neg b\} S_2 \{Q\} \\
\{P\} \text{ if } b \text{ then } S_1 \text{ else } S_2 \{Q\}
\]

ROI for *if_then* statement:

\[
\{P \land b\} S \{Q\}, (P \land \neg b) \Rightarrow Q \\
\{P\} \text{ if } b \text{ then } S \{Q\}
\]
Reasoning about Iteration

• Consider the assertion: \( \{P\} \text{ while } b \text{ do } S \{Q\} \)

• What are the necessary conditions for this assertion to hold?

• What are the *necessary* conditions for this assertion to hold?
Consider a Loop “Invariant” - I

Suppose I holds initially...

is preserved by S...

and implies Q when and if the loop finally terminates...

then the assertion would hold!
Sufficient Conditions: while_do

• Thus, a ROI for the `while_do` statement is:

\[
P \Rightarrow I, \ {I \land b} \ S \ {I}, \ (I \land \neg b) \Rightarrow Q \\
\{P\} \ while \ b \ do \ S \ \{Q\}
\]

where the three antecedents are sometimes given the names `initialization`, `preservation`, and `finalization`, respectively.
Use the invariant $I: Z=XJ$ to prove:

\begin{align*}
\{\text{true}\} & \quad \text{Initialization: } P \Rightarrow I \\
Z := X & \quad \text{Preservation: } \{I \land b\} S \{I\} \\
J := 1 & \quad \text{Finalization: } (I \land \neg b) \Rightarrow Q \\
\text{while } J<>Y \text{ do} & \\
Z := Z+X & \\
J := J+1 & \\
\text{end_while} & \\
\{Z=XY\}
\end{align*}
Example 3

Use the invariant $I: Z=XJ$ to prove:

$$\{\text{true}\} \quad \text{Initialization: } P \Rightarrow I$$

$$Z := X \quad \text{What is } \text{“}P\text{”}?$$

$$J := 1$$

$$\text{while } J<>Y \text{ do}$$

$$Z := Z+X$$

$$J := J+1$$

$$\text{end}_\text{while}$$

$$\{Z=XY\}$$
Example 3

Use the invariant $I: Z=XJ$ to prove:

\[
\{\text{true}\} \quad \text{Initialization: } P \Rightarrow I
\]

\[
\begin{align*}
Z & := X \\
J & := 1
\end{align*}
\]

$P \quad \text{while } J<>Y \text{ do}$

\[
\begin{align*}
Z & := Z+X \\
J & := J+1
\end{align*}
\]

$\text{end}_\text{while}$

\[
\{Z=XY\}
\]

What is “$P$”?
Example 3

Use the invariant I: Z=XJ to prove:

\{\text{true}\} \quad \text{Initialization: } P \Rightarrow I

\begin{align*}
Z &:= X \\
J &:= 1 \\
\text{while } J \neq Y \text{ do} \\
\quad Z &:= Z + X \\
\quad J &:= J + 1 \\
\text{end\_while}
\end{align*}

\{Z=XY\}

What is “P”?

\(Z=X \land J=1\)
Use the invariant $I: Z=XJ$ to prove:

$\{\text{true}\}$

- $Z := X$
- $J := 1$

$P \rightarrow$ while $J<>Y$ do
  - $Z := Z+X$
  - $J := J+1$
end\_while

$\{Z=XY\}$

**Initialization:** $P \Rightarrow I$

What is “$P$”?

$(Z=X \land J=1)$

Does $(Z=X \land J=1) \Rightarrow Z=XJ$?
Example 3

Use the invariant $I: Z=XJ$ to prove:

$$\{\text{true}\}$$

Initialization: $P \Rightarrow I$

What is “P”?

$$(Z=X \land J=1)$$

Does $(Z=X \land J=1) \Rightarrow Z=XJ$?

Yep!

$P$ →

while $J<>Y$ do

$$Z := Z+X$$

$$J := J+1$$

end_while

$$\{Z=XY\}$$
Example 3

Use the invariant $I: Z = XJ$ to prove:

\[
\begin{align*}
\{\text{true}\} & \quad \textbf{Initialization: } P \implies I \checkmark \\
Z & := X \\
J & := 1 \\
\text{while } J \neq Y \text{ do} \\
& \quad Z := Z + X \\
& \quad J := J + 1 \\
\text{end\_while} \\
\{Z = XY\}
\end{align*}
\]
Example 3

Use the invariant $I: Z=XJ$ to prove:

\[
\{\text{true}\} \quad \text{Initialization: } P \Rightarrow I \checkmark
\]

\[
Z := X \\
J := 1 \\
\text{while } J<>Y \text{ do} \\
\quad Z := Z+X \\
\quad J := J+1 \\
\text{end_while}
\]

\[
\{Z=XY\}
\]

\[
\text{Preservation: } \{I \land b\} \mathcal{S} \{I\}
\]
Example 3

Use the invariant $I: Z= XJ$ to prove:

\[
\{\text{true}\} \quad \text{Initialization: } P \Rightarrow I \checkmark
\]

\[
Z := X \\
J := 1 \\
\text{while } J<>Y \text{ do} \\
\quad Z := Z+X \\
\quad J := J+1 \\
\text{end}_\text{while}
\]

\[
\{Z=XY\} \\
\text{Preservation: } \{I \land b\} S \{I\}
\]

What are “b” and “S”?
Example 3

Use the invariant $I: Z=XJ$ to prove:

\[
\begin{align*}
\{\text{true}\} & & \text{Initialization: } P \Rightarrow I \checkmark \\
Z := X & & \text{Preservation: } \{I \land b\} S \{I\} \\
J := 1 & \\
\text{while } J<>Y \text{ do} & \\
& \begin{cases} 
Z := Z+X \\
J := J+1 
\end{cases} \downarrow S \\
\text{end}_\text{while} & \\
\{Z=XY\}
\end{align*}
\]
**Example 3**

Use the invariant $I: Z=XJ$ to prove:

\[
\begin{align*}
\{\text{true}\} & \\
\{Z=XJ \land J\neq Y\} & \quad \text{Preservation: } \{I \land b\} S \{I\}
\end{align*}
\]

\[
\begin{align*}
Z & := X \quad b \\
J & := 1 \\
\text{while } J\neq Y \text{ do} & \\
Z & := Z+X \\
J & := J+1 \\
\text{end}\_while
\end{align*}
\]

\[
\{Z=XY\}
\]

Initialization: $P \Rightarrow I \checkmark$
Example 3

Use the invariant $I: Z=XJ$ to prove:

$$\{\text{true}\}$$

$Z := X\quad b$
$J := 1$

while $J<>Y$ do

$Z := Z+X$
$J := J+1$

end_while

$$\{Z=XY\}$$

Initialization: $P \Rightarrow I \checkmark$

Preservation: $\{I \land b\} \implies \{I\}$

$\{Z=XJ \land J\neq Y\}$
$Z := Z+X$
Example 3

Use the invariant I: $Z = XJ$ to prove:

\[
\{\text{true}\} \\
Z := X \quad b \\
J := 1 \\
\text{while } J \neq Y \text{ do} \\
  Z := Z + X \\
  J := J + 1 \\
\text{end_while} \\
\{Z = XY\}
\]

**Initialization:** $P \Rightarrow I \checkmark$

**Preservation:** $\{I \land b\} S \{I\}$

- $\{Z = XJ \land J \neq Y\}$
- $Z := Z + X$
- $\{Z = X(J+1) \land \ldots\}$
Example 3

Use the invariant $I: Z=XJ$ to prove:

\[
\{\text{true}\} \\
Z := X \\
J := 1 \\
\text{while } J \neq Y \text{ do} \\
\quad Z := Z + X \\
\quad J := J + 1 \\
\text{end}_\text{while} \\
\{Z=XY\}
\]

**Initialization:** $P \Rightarrow I \checkmark$

**Preservation:** $\{I \land b\} S \{I\}$

$\{Z=XJ \land J \neq Y\}$

$\{Z := Z + X\}$

$\{Z = X(J+1) \land J \neq Y\}$
Example 3

Use the invariant $I: Z=XJ$ to prove:

$$\{\text{true}\}$$

$Z := X$
$J := 1$
while $J<>Y$ do
  $Z := Z+X$
  $J := J+1$
end_while

$\{Z=XY\}$

Initialization: $P \Rightarrow I \checkmark$

Preservation: $\{I \land b\} S \{I\}$

$$\{Z=XJ \land J\neq Y\}$$
$$Z := Z+X$$
$$\{Z=X(J+1) \land J\neq Y\}$$
$$J := J+1$$
Example 3

Use the invariant $I: Z=XJ$ to prove:

$$
\{\text{true}\} \\
Z := X \quad b \\
J := 1 \\
\text{while } J<>Y \text{ do} \\
\quad Z := Z+X \\
\quad J := J+1 \\
\text{end\_while} \\
\{Z=XY\}
$$

Initialization: $P \Rightarrow I \checkmark$

Preservation: $\{I \land b\} S \{I\}$

$\{Z=XJ \land J\neq Y\}$

$Z := Z+X$

$\{Z=X(J+1) \land J\neq Y\}$

$J := J+1$
Example 3

Use the invariant \( I: Z=XJ \) to prove:

\[
\{ \text{true} \} \\
\{ Z := X \} \quad \text{b} \\
\{ J := 1 \} \\
\{ \text{while } J<>Y \text{ do} \} \\
\{ Z := Z+X \} \\
\{ J := J+1 \} \\
\{ \text{end\_while} \} \\
\{ Z=X\text{Y} \}
\]

**Initialization:** \( P \Rightarrow I \checkmark \\
**Preservation:** \( \{ I \wedge b \} S \{ I \} \\
\{ Z=XJ \wedge J\neq Y \} \\
\{ Z := Z+X \} \\
\{ Z=X(J+1) \wedge J\neq Y \} \\
\{ J := J+1 \} \\
\{ Z=X((J-1)+1) \wedge J-1\neq Y \} \)
Example 3

Use the invariant I: Z=XJ to prove:

\{
\text{true}
\}

\begin{align*}
\text{Initialization: } & P \implies I \checkmark \\
\text{Preservation: } & \{I \land b\} \ S \ {\{I\}} \\
\{Z=XJ \land J \neq Y\} & \\
Z & := Z+X \\
J & := J+1 \\
\{Z=X((J-1)+1) \land J-1 \neq Y\} & \\
\implies Z=XJ
\end{align*}

\begin{align*}
Z := X \\
J := 1 \\
\text{while } J \not\equiv Y \text{ do} \\
Z := Z+X \\
J := J+1 \\
\text{end}_\text{while}
\end{align*}
Example 3

Use the invariant $I: Z=XJ$ to prove:

\[
\begin{align*}
\{\text{true}\} & \quad \text{Initialization: } P \Rightarrow I \checkmark \\
Z := X & \quad \text{Preservation: } \{I \land b\} \subseteq \{I\} \checkmark \\
J := 1 & \\
\text{while } J \neq Y \text{ do} & \\
Z := Z+X & \\
J := J+1 & \\
\text{end}_\text{while} & \\
\{Z=XY\}
\end{align*}
\]
Example 3

Use the invariant \( I: Z=XJ \) to prove:

\[
\{ \text{true} \} \\
Z := X \\
J := 1 \\
\text{while } J \ne Y \text{ do} \\
\quad Z := Z+X \\
\quad J := J+1 \\
\text{end\_while} \\
\{ Z=XY \}
\]

**Initialization:** \( P \Rightarrow I \checkmark \)

**Preservation:** \( \{ I \land b \} \ S \{ I \} \checkmark \)

**Finalization:** \( (I \land \neg b) \Rightarrow Q \)
Example 3

Use the invariant $I: Z=XJ$ to prove:

\[
\{\text{true}\}
\]

\[
\begin{align*}
Z &:= X \\
J &:= 1 \\
\text{while } J &\not\equiv Y \text{ do} \\
Z &:= Z+X \\
J &:= J+1 \\
\text{end_while}
\end{align*}
\]

\[
\{Z=XY\}
\]

**Initialization:** $P \Rightarrow I \checkmark$

**Preservation:** $\{I \land b\} S \{I\} \checkmark$

**Finalization:** $(I \land \neg b) \Rightarrow Q$

Does $(Z=XJ \land J=Y) \Rightarrow Z=XY$?
Example 3

Use the invariant $I: Z=XJ$ to prove:

{true} 

\[
\begin{align*}
Z &:= X \\
J &:= 1 \\
\text{while } J&<>Y \text{ do} \\
Z &:= Z+X \\
J &:= J+1 \\
\text{end_while}
\end{align*}
\]

{Z=XY}

**Initialization:** $P \Rightarrow I \checkmark$

**Preservation:** $\{I \land b\} \Rightarrow \{I\} \checkmark$

**Finalization:** $(I \land \neg b) \Rightarrow Q$

Does $(Z=XJ \land J=Y) \Rightarrow Z=XY$?

Yep!
Example 3

Use the invariant $I: Z=XJ$ to prove:

\[
\{\text{true}\} \quad \text{Initialization: } P \Rightarrow I \; \checkmark
\]

\[
Z := X \\
J := 1 \\
\text{while } J \not<\!> Y \text{ do} \\
\quad Z := Z + X \\
\quad J := J + 1 \\
\text{end\_while}
\]

\[
\{Z=XY\} \\
\text{Preservation: } \{I \land b\} \Rightarrow \{I\} \; \checkmark
\]

\[
\text{Finalization: } (I \land \neg b) \Rightarrow Q \; \checkmark
\]
Exercise

- See **WHILE LOOP VERIFICATION EXERCISE** on course website
Some Limitations of Formal Verification

- Difficulties can arise when dealing with:
  - parameters
  - pointers
  - synthesis of invariants
  - decidability of verification conditions
  - concurrency
Some Limitations of Formal Verification (cont’d)

• In addition, a formal specification:
  – may be expensive to produce
  – may be incorrect and/or incomplete
  – normally reflects *functional* requirements only

• Will the proof process be manual or automatic? Who will prove the proof?
That’s all, folks, but if you like formal verification…

- Take CEN 6070, Software Testing & Verification and learn about:
  - deriving invariants using the Invariant Status Theorem,
  - proving termination using the Method of Well-Founded Sets,
  - Predicate transforms ("weakest pre-conditions")
  - function-theoretic verification (prove the correctness of loops without invariants!)
  - and MUCH more!
Proofs of Correctness: An Introduction to Axiomatic Verification

CEN 5035
Software Engineering

Prepared by
Stephen M. Thebaut, Ph.D.
University of Florida