Proofs and Proof Outlines for Partial Correctness

Part 1: Full Proofs and Proof Outlines of Partial Correctness

CS 536: Science of Programming, Fall 2021

A. Why

• A formal proof lets us write out in detail the reasons for believing that something is valid.

B. Objectives

At the end of this activity assignment you should be able to

• Write and check formal proofs of partial correctness.
• Translate between full formal proofs and full proof outlines

C. Problems

Formal Proofs

1. The formal Hilbert-style proof below is incomplete; fill in the missing rule names (and line references, where needed).
   1. \( T \rightarrow 0 \geq 0 \land 1 = 2^0 \) __________
   2. \( \{ 0 \geq 0 \land 1 = 2^0 \} \ k := 0 \ \{ k \geq 0 \land 1 = 2^k \} \) __________
   3. \( \{ T \} \ k := 0 \ \{ k \geq 0 \land 1 = 2^k \} \) __________
   4. \( \{ k \geq 0 \land 1 = 2^k \} \ x := 1 \ \{ k \geq 0 \land x = 2^k \} \) __________
   5. \( \{ T \} \ k := 0; \ x := 1 \ \{ k \geq 0 \land x = 2^k \} \) __________

Here’s an alternate version of the proof that uses forward assignments:

1. \( \{ T \} \ k := 0 \ \{ k = 0 \} \) __________
2. \( \{ k = 0 \} \ x := 1 \ \{ k = 0 \land x = 1 \} \) __________
3. \( \{ T \} \ k := 0; \ x := 1 \ \{ k = 0 \land x = 1 \} \) __________
4. \( k = 0 \land x = 1 \rightarrow k \geq 0 \land x = 2^k \) __________
5. \( \{ T \} \ k := 0; \ x := 1 \ \{ k \geq 0 \land x = 2^k \} \) __________

2. Repeat Problem 1 on the incomplete proof below.

1. \( \{ -x \ = \ \text{abs}(x) \} \ y := -x \ \{ y \ = \ \text{abs}(x) \} \) __________
2. \( \ y = x \land x < 0 \rightarrow -x = \text{abs}(x) \) __________
3. \( \{ y = x \land x < 0 \} \ y := -x \ \{ y = \text{abs}(x) \} \) __________
4. \( \{ y = \text{abs}(x) \} \ \text{skip} \ \{ y = \text{abs}(x) \} \) __________
5. \( y = x \land x \geq 0 \rightarrow y = \text{abs}(x) \) __________
6. \{ y = x \land x \geq 0 \} \text{skip} \{ y = \text{abs}(x) \} \\
7. \{ y = x \} \text{if } x < 0 \text{ then } y := -x \text{ fi } \{ y = \text{abs}(x) \}

3. Repeat Problem 1 on the incomplete proof below.

Below, let \( W \equiv \text{while } j > 0 \text{ do } j := j-1; s := s+j \text{ od} \) [and \( p \equiv 0 \leq j \leq n \land s = \text{sum}(j, n) \)] added 10/19

\begin{enumerate}
\item \( \{ p[n/s] \} s := n \{ p \} \) assignment (backwards)
\item \( \{ p[n/s][n/j] \} j := n \{ p[n/s] \} \) assignment (backwards)
\item \( \{ p[n/s] \} j := n \{ p \} \) sequence 2, 1
\item \( n \geq 0 \rightarrow p[n/s][n/j] \) predicate logic
\item \( \{ n \geq 0 \} j := n; s := n \{ p \} \) precondition strength. 4, 3
\item \( \{ p \land j > 0 \} j := j-1 \{ p1 \} \) assignment (forwards)
\begin{itemize}
\item where \( p1 \equiv p[j_0/j] \land j = j_0-1 \)
\end{itemize}
\item \( \{ p2 \} s := s+j \{ p2 \} \) assignment (forwards)
\begin{itemize}
\item where \( p2 \equiv p2[s_0/s] \land s = s_0 + j \)
\end{itemize}
\item \( \{ p \land j > 0 \} j := j-1; s := s+j \{ p2 \} \) sequence 6, 7
\item \( p2 \rightarrow p \) predicate logic
\item \( \{ p \land j > 0 \} j := j-1; s := s+j \{ p \} \) postcondition weak. 8, 9
\item \( \{ \text{inv} p \} W \{ p \land j \leq 0 \} \) while 10
\item \( \{ n \geq 0 \} j := n; s := n \{ \text{inv} p \} W \{ p \land j \leq 0 \} \text{sequence 5, 11} \)
\item \( p \land j \leq 0 \rightarrow s = \text{sum}(0, n) \) predicate logic
\item \( \{ n \geq 0 \} j := n; s := n \{ \text{inv} p \} W \{ s = \text{sum}(0, n) \} \) postcondition weak.
\item \( 12, 13 \)
\end{enumerate}

**Full Proof Outlines**

For Problems 1–3, you are given a full proof outline; write a corresponding Hilbert-style proof of partial correctness from it. There are multiple right answers.

1. \( \{ T \} \{ 0 \geq 0 \land 1 = 2^0 \} k := 0; \{ k \geq 0 \land 1 = 2^k \} x := 1 \{ k \geq 0 \land x = 2^k \} \)

2a. \( \{ y = x \} \text{if } x < 0 \text{ then} \)
\begin{itemize}
\item \( \{ y = x \land x < 0 \} \{ -x = \text{abs}(x) \} y := -x \{ y = \text{abs}(x) \} \)
\item else \( \{ y = x \land x \geq 0 \} \{ y = \text{abs}(x) \} \text{skip} \{ y = \text{abs}(x) \} \)
\item fi \( \{ y = \text{abs}(x) \} \)
\end{itemize}
2b. \( \{ y = x \} \) if \( x < 0 \) then
   \( \{ y = x \land x < 0 \} \ y := -x \ \{ y_0 = x \land x < 0 \land y = -x \} \)
else
   \( \{ y = x \land x \geq 0 \} \) skip \( \{ y = x \land x \geq 0 \} \)
fi \( \{ (y_0 = x \land x < 0 \land y = -x) \lor (y = x \land x \geq 0) \} \) \( \{ y = \text{abs}(x) \} \)

2c. \( \{ y = x \} \) \( \{(x < 0 \rightarrow -x = \text{abs}(x)) \land (x \geq 0 \rightarrow y = \text{abs}(x))\} \)
if \( x < 0 \) then
   \( \{-x = \text{abs}(x)\} \) \( y := -x \ \{ y = \text{abs}(x) \} \)
else
   \( \{ y = \text{abs}(x) \} \) skip \( \{ y = \text{abs}(x) \} \)
fi \( \{ y = \text{abs}(x) \} \)

3. Hint: Use \( \text{sp} \) for the two loop initialization assignments.
\( \{ n \geq 0 \} \ j := n; \ \{ n \geq 0 \land j = n \} \ s := n; \ \{ n \geq 0 \land j = n \land s = n \} \)
\( \{ \text{inv } p \equiv 0 \leq j \leq n \land s = \text{sum}(j, n) \} \)
while \( j > 0 \) do
   \( \{ p \land j > 0 \} \ \{ p[s+j/s][j-1/j] \} \ j := j-1 \)
   \( \{ p[s+j/s] \} \ s := s+j \ \{ p \} \)
   od
\( \{ p \land j \leq 0 \} \ \{ s = \text{sum}(0, n) \} \)
Solution to Practice 16 (Formal Proofs and Full Proof Outlines)

1. Proof:
   1. \( T \rightarrow 0 \geq 0 \land 1 = 2^0 \)  
      predicate logic
   2. \( \{ 0 \geq 0 \land 1 = 2^0 \} \ k := 0 \ \{ k \geq 0 \land 1 = 2^k \} \)  
      assignment (backward)
   3. \( \{ k \geq 0 \land 1 = 2^k \} \ x := 1 \ \{ k \geq 0 \land x = 2^k \} \)  
      assignment (backward)
   4. \( \{ k \geq 0 \land x = 2^k \} \ x := 1 \ \{ k \geq 0 \land x = 2^k \} \)  
      sequence 3, 4
   5. \( \{ T \} \ k := 0; x := 1 \ \{ k \geq 0 \land x = 2^k \} \)  
      postcondition weakening 3, 4

   [Alternate version]
   1. \( \{ T \} \ k := 0 \ \{ k = 0 \} \)  
      assignment (forward)
   2. \( \{ k = 0 \} \ x := 1 \ \{ k = 0 \land x = 1 \} \)  
      assignment (forward)
   3. \( \{ k = 0 \land x = 1 \} \ x := 1 \ \{ k = 0 \land x = 1 \} \)  
      sequence 1, 2
   4. \( \{ k = 0 \land x = 1 \ \land k \geq 0 \land x = 2^k \} \)  
      predicate logic
   5. \( \{ k = 0 \land x = 1 \} \ x := 1 \ \{ k \geq 0 \land x = 2^k \} \)  
      sequence 3, 4

2. Proof:
   1. \( \{ -x = abs(x) \} \ y := -x \ \{ y = abs(x) \} \)  
      assignment (backward)
   2. \( \{ y = x \land x < 0 \rightarrow -x = abs(x) \} \)  
      predicate logic
   3. \( \{ y = x \land x < 0 \} \ y := -x \ \{ y = abs(x) \} \)  
      precondition strength. 2, 1
   4. \( \{ y = abs(x) \} \ skip \ \{ y = abs(x) \} \)  
      skip
   5. \( \{ y = x \land x \geq 0 \} \ y := -x \ \{ y = abs(x) \} \)  
      precondition strength. 5, 4
   7. \( \{ y = x \} \ if \ x < 0 \ then \ y := -x \ fi \ \{ y = abs(x) \} \)  
      conditional 3, 6

3. Below, \( W \equiv while \ j > 0 \ do \ j := j-1; \ s := s+j \ od \)
   1. \( \{ n \geq 0 \} \ j := n \ \{ n \geq 0 \land j = n \} \)  
      assignment (forward)
   2. \( \{ n \geq 0 \land j = n \} \ s := n \ \{ n \geq 0 \land j = n \land s = n \} \)  
      assignment (forward)
   3. \( \{ n \geq 0 \land j = n \land s = n \} \ j := n \ \{ n \geq 0 \land j = n \land s = n \} \)  
      sequence 1, 2
   4. \( n \geq 0 \land j = n \land s = n \rightarrow p \)  
      predicate logic
   5. \( \{ n \geq 0 \} \ j := n; \ s := n \ \{ p \} \)  
      postcondition weak. 3, 4
   6. \( \{ p[s+j/s] \} \ s := s+j \ \{ p \} \)  
      assignment (backward)
   7. \( \{ p[s+j/s][j-1/j] \} \ j := j-1 \ \{ p[s+j/s] \} \)  
      assignment (backward)
   8. \( p \land j > 0 \rightarrow p[s+j/s][j-1/j] \)  
      predicate logic
   9. \( \{ p \land j > 0 \} \ j := j-1 \ \{ p[s+j/s] \} \)  
      precondition strength. 8, 7
  10. \( \{ p \land j > 0 \} \ j := j-1; \ s := s+j \ \{ p \} \)  
      sequence 9, 6
  11. \( \{ inv \ p \} \ W \ \{ p \land j \leq 0 \} \)  
      while 10
  12. \( p \land j \leq 0 \rightarrow s = sum(0, n) \)  
      predicate logic
  13. \( \{ inv \ p \} \ W \ \{ s = sum(0, n) \} \)  
      postcondition weak. 12, 11
  14. \( \{ n \geq 0 \} \ j := n; \ s := n; \ \{ inv \ p \} \ W \ \{ s = sum(0, n) \} \)  
      sequence 5, 13
Full Proof Outlines (Solution)

1. (Full outline to proof):
   1. $T \rightarrow 0 \geq 0 \land 1 = 2^0$  
      predicate logic
   2. $\{0 \geq 0 \land 1 = 2^0\} \ k := 0; \ {k \geq 0 \land 1 = 2^k}$  
      assignment (backwards)
   3. $\{T\} \ k := 0; \ {k \geq 0 \land 1 = 2^k}$  
      precondition strength. 1, 2
   4. $\{k \geq 0 \land 1 = 2^k\} \ x := 1 \ {k \geq 0 \land x = 2^k}$  
      assignment (backwards)
   5. $\{T\} \ k := 0; \ x := 1 \ {k \geq 0 \land x = 2^k}$  
      sequence 3, 4

2a. (Full outline to proof):
   1. $\{-x = \text{abs}(x)\} \ y := -x \ {y = \text{abs}(x)}$  
      assignment (backwards)
   2. $y = x \land x < 0 \rightarrow -x = \text{abs}(x)$  
      predicate logic
   3. $\{y = x \land x < 0\} \ y := -x \ {y = \text{abs}(x)}$  
      precondition strength. 2, 1
   4. $\{y = \text{abs}(x)\} \ \text{skip} \ {y = \text{abs}(x)}$  
      skip
   5. $y = x \land x \geq 0 \rightarrow y = \text{abs}(x)$  
      predicate logic
   6. $\{y = x \land x \geq 0\} \ \text{skip} \ {y = \text{abs}(x)}$  
      precondition strength. 5, 4
   7. $\{y = x\} \ \text{if} \ x < 0 \ \text{then} \ y := -x \ \text{fi} \ {y = \text{abs}(x)}$  
      conditional 3, 6

2b. (Full outline to proof):
   1. $\{y = x \land x < 0\} \ y := -x \ {y_0 = x \land x < 0 \land y = -x}$  
      assignment (forward)
   2. $\{y = x \land x \geq 0\} \ \text{skip} \ {y = x \land x \geq 0}$  
      skip
   3. $\{y = x\} \ \text{if} \ x < 0 \ \text{then} \ y := -x \ \text{fi}$  
      $\{y_0 = x \land x < 0 \land y = -x\} \lor \{y = x \land x \geq 0\}$  
      conditional 1, 2
   4. $(y_0 = x \land x < 0 \land y = -x) \lor \{y = x \land x \geq 0\} \rightarrow y = \text{abs}(x)$  
      predicate logic
   5. $\{y = x\} \ \text{if} \ x < 0 \ \text{then} \ y := -x \ \text{fi} \ {y = \text{abs}(x)}$  
      postcondition weak., 3, 4

2c. (Full outline to proof):
   1. $\{-x = \text{abs}(x)\} \ y := -x \ {y = \text{abs}(x)}$  
      assignment (backwards)
   2. $\{y = \text{abs}(x)\} \ \text{skip} \ {y = \text{abs}(x)}$  
      skip
   3. $\{p\} \ \text{if} \ x < 0 \ \text{then} \ y := -x \ \text{fi} \ {y = \text{abs}(x)}$  
      conditional 1, 2
      where $p \equiv (x < 0 \rightarrow -x = \text{abs}(x)) \land (x \geq 0 \rightarrow y = \text{abs}(x))$
   4. $y = x \rightarrow p$  
      predicate Logic
   5. $\{y = x\} \ \text{if} \ x < 0 \ \text{then} \ y := -x \ \text{fi} \ {y = \text{abs}(x)}$  
      precondition strength. 4, 3
3. Below, let $W \equiv \text{while } k > 0 \text{ do } k := k - 1; s := s + k \text{ od}$

1. $\{ n \geq 0 \} k := n \{ n \geq 0 \land k = n \}$

2. $\{ n \geq 0 \land k = n \} s := n \{ n \geq 0 \land k = n \land s = n \}$

3. $\{ n \geq 0 \} k := n; s := n \{ n \geq 0 \land k = n \land s = n \}$

4. $n \geq 0 \land k = n \land s = n \rightarrow p$

5. $\{ n \geq 0 \} k := n; s := n \{ p \}$

6. $\{ p[s+k/s] \} s := s + k \{ p \}$

7. $\{ p[s+k/s][k-1/k] \} k := k - 1 \{ p[s+k/s] \}$

8. $p \land k > 0 \rightarrow p[s+k/s][k-1/k]$

9. $\{ p \land k > 0 \} k := k - 1 \{ p[s+k/s] \}$

10. $\{ p \land k > 0 \} k := k - 1; s := s + k \{ p \}$

11. $\{ inv \ p \} W \{ p \land k \leq 0 \}$

12. $p \land k \leq 0 \rightarrow s = \text{sum}(0, n)$

13. $\{ inv \ p \} W \{ s = \text{sum}(0, n) \}$

14. $\{ n \geq 0 \} k := n; s := n; W \{ s = \text{sum}(0, n) \}$