Total correctness example - calculating the nth power of a natural number

In this section we can present an additional example for already presented notions like deadlock freedom, interference freedom and verification of (sequential) programs.

1 Problem

Given the following problem:
$A = (x{:}\mathbb{N}, n{:}\mathbb{N}, z{:}\mathbb{N})$

$Pre = (x = x' \wedge n = n' \wedge x > 0)$

$Post = (z = x'^{n'})$

Prove the total correctness of the following (annotated) program (let us denote it by S) with respect to the given problem.

```
{$x = x' \wedge n = n' \wedge x > 0$}
$z := 1$;
{$Inv$}
parbegin  $S_1 \| S_2$  parend
{$z = x'^{n'}$}
```

```
$S_1$:

{$Inv$}
while  $n \neq 0$  do
   {$Inv \wedge n \neq 0$}
   n,z  :=  n−1,z·x
od
{$z = x'^{n'} \wedge n = 0$}
```

```
$S_2$:

{$Inv$}
while  $n \neq 0$  do
   {$Inv$}
   await  even(n)  then
      x,n  :=  x·x,n/2
   ta
od
{$z = x'^{n'} \wedge n = 0$}
```

Inv denotes the invariant of the loops and is given as follows: $Inv = (z \cdot x^n = x'^{n'})$
Variant function of the loops: t: n

Verification conditions

Notice that S is a sequence of two statements:

$\{x = x' \wedge n = n' \wedge x > 0\}$
$\{x > 0\}$
$z := 1;$
$\{Inv\}$
T: **parbegin** $S_1 \| S_2$ **parend**
$\{z = x'^{n'}\}$

In order to prove the total correctness of S with respect to the given problem, we have to prove that S takes us from precondition Pre to postcondition $Post$. Since S is a sequence, that is

- $(x = x' \wedge n = n' \wedge x > 0) \Rightarrow wp(z := 1, Inv)$
- The parallel block (let us denote it by T) takes us from the intermediate Inv to the postcondition $z = x'^{n'}$
  - The components S_1 and S_2 are totally correct with respect to their specification: $pre(S_1) \Rightarrow wp(S_1, post(S_1))$ and $pre(S_2) \Rightarrow wp(S_2, post(S_2))$
  - Entry condition: When the precondition of T holds, then any of the two components is able to be executed. In other words both the precondition of S_1 and the precondition of S_2 hold: $pre(T) \Rightarrow pre(S_1) \wedge pre(S_2)$
  - Exit condition: When all components terminate by reaching their respective postcondition, then the program T also reaches its expected postcondition: $post(S_1) \wedge post(S_2) \Rightarrow post(T)$
  - The parallel block T is free from deadlock.
  - The interference freedom holds; S_1 and S_2 do not invalidate each other's behaviour.

1.1 Proof - 1st part of the sequence

We want to prove that $(x = x' \wedge n = n' \wedge x > 0) \Rightarrow wp(z := 1, Inv)$ holds. $wp(z := 1, Inv) = Inv^{z \leftarrow 1} = (1 \cdot x^n = x'^{n'})$, it obviously can be deduced from $(x = x' \wedge n = n' \wedge x > 0)$

1.2 Proof - 2nd part of the sequence - The components are totally correct

Proof - 2nd part of the sequence - Components of T are totally correct - S1 is totally correct

We want to prove that $pre(S_1) \Rightarrow wp(S_1, post(S_1))$. Since S_1 is a loop, by the verification rule of the loop it is sufficient to prove the followings:

- $pre(S_1) \Rightarrow Inv$ obviously holds since the precondition of S_1 is the invariant itself.

- $(Inv \wedge \neg(n \neq 0)) \Rightarrow post(S_1)$
  $(z = x'^{n'} \wedge n = 0) \Rightarrow (z = x'^{n'} \wedge n = 0)$ obviously holds.

- $(Inv \wedge n \neq 0) \Rightarrow t > 0$
  Our variant function is n, that is a natural number or zero ($n : \mathbb{N}$). As $n \neq 0$, $n > 0$ holds.

- $(Inv \wedge n \neq 0 \wedge t = t_0) \Rightarrow wp(n, z := n-1, z \cdot x, Inv \wedge t < t_0)$ that is
  $(Inv \wedge n \neq 0 \wedge n = t_0) \Rightarrow wp(n, z := n-1, z \cdot x, Inv \wedge n < t_0)$
  Let us calculate the weakest precondition $wp(n, z := n-1, z \cdot x, Inv \wedge n < t_0)$!
  $wp(n, z := n-1, z \cdot x, Inv \wedge n < t_0) =$
  $(Inv^{n \leftarrow n-1, z \leftarrow z \cdot x} \wedge n-1 \in \mathbb{N}_0 \wedge n-1 < t_0) = (z \cdot x \cdot x^{n-1} = x'^{n'} \wedge n-1 \in \mathbb{N} \wedge n-1 < t_0)$

  - $z \cdot x \cdot x^{n-1} = x'^{n'}$
    This holds since $(z \cdot x^n = x'^{n'}$ holds by the invariant.
  - $n-1 \in \mathbb{N}$
    We know that n is a natural number or zero ($n : \mathbb{N}$). Moreover, by the loopcondition $n > 0$, that is $n \geq 1$. Then $n-1 \geq 0$ holds.
  - $n-1 < t_0$
    This obviously holds since $n = t_0$

Proof - 2nd part of the sequence - Components of T are totally correct - S2 is totally correct

We want to prove that $pre(S_2) \Rightarrow wp(S_2, post(S_2))$. Since S_2 is a loop, by the verification rule of the loop it is sufficient to prove the followings:

- $pre(S_2) \Rightarrow Inv$ obviously holds since the precondition of S_2 is the invariant itself.

- $(Inv \wedge \neg(n \neq 0)) \Rightarrow post(S_2)$
  $(z = x'^{n'} \wedge n = 0) \Rightarrow (z = x'^{n'} \wedge n = 0)$ obviously holds.

- $(Inv \wedge n \neq 0) \Rightarrow t > 0$
  Our variant function is n that is a natural number or zero ($n : \mathbb{N}$). As $n \neq 0$, $n > 0$ holds.

- $(Inv \wedge n \neq 0 \wedge t = t_0) \Rightarrow wp($**await even(n) then** $x, n := x \cdot x, n/2$ **ta**$, Inv \wedge t < t_0)$.
  Recall the verification rule of the await statement: If

  1. $Q \implies \beta \vee \neg\beta$ and
  2. $Q \wedge \beta \implies wp(S, R)$

  then $Q \implies wp(S, R)$ holds. By the rule of the await statement it is sufficient to prove that

  1. $(Inv \wedge n \neq 0 \wedge t = t_0 \implies even(n) \vee \neg even(n))$. This holds, as n is a natural number, it is either even or odd.

2. $(Inv \land n \neq 0 \land t = t_0 \land even(n)) \Rightarrow wp(x, n := x \cdot x, n/2, Inv \land t < t_0)$. That is $(Inv \land n \neq 0 \land n = t_0 \land even(n)) \Rightarrow wp(x, n := x \cdot x, n/2, Inv \land n < t_0)$

Let us calculate the weakest precondition!
$(Inv \land n \neq 0 \land n = t_0 \land even(n)) \Rightarrow wp(x, n := x \cdot x, n/2, Inv \land n < t_0) =$
$(Inv^{x \leftarrow x \cdot x, n \leftarrow n/2} \land even(n) \land n/2 < t_0) = (z \cdot (x \cdot x)^{n/2} = x'^{n'} \land even(n) \land n/2 < t_0)$

- $z \cdot (x \cdot x)^{n/2} = x'^{n'}$
  This holds since $z \cdot x^n = x'^{n'}$ holds due to the invariant.
- $even(n)$
  We know that n is an even number. Note that this condition guarantees that the program $n := n/2$ does not abort.
- $n/2 < t_0$
  This obviously holds since $n = t_0$ and $n \neq 0$ (n:$\mathbb{N}$). When we divide a positive number by two, it will be less.

Proof - 2nd part of the sequence - Entry/Exit conditions hold

$\{Inv\}$
T: **parbegin** $S_1 \| S_2$ **parend**
$\{z = x'^{n'}\}$

- When the precondition of T holds, then any of the two components is able to be executed. In other words, if $pre(T)$ holds then both the precondition of S_1 and the precondition of S_2 hold:
  $pre(T) \Rightarrow (pre(S_1) \land pre(S_2))$

  Proof:
  We know that $pre(T) = Inv$ and $pre(S_1) = pre(S_2) = Inv$, these assertions are given.
  Now $Inv \Rightarrow (Inv \land Inv)$ obviously holds.

- When all components terminate by reaching their respective postcondition, then the program T also reaches its expected postcondition: $(post(S_1) \land post(S_2)) \Rightarrow post(T)$

  Proof:
  We know that $post(T) = (z = x'^{n'})$ and $post(S_1) = post(S_2) = (z = x'^{n'} \land n = 0)$, these assertions are given.
  Now $((z = x'^{n'} \land n = 0) \land (z = x'^{n'} \land n = 0)) \Rightarrow z = x'^{n'}$ obviously holds.

Proof - 2nd part of the sequence - Deadlock freedom

Deadlock can only occur when the program has to wait at an **await** statement. Now S contains only two statements, the first one is an assignment $z := 1$, the other one is a **parbegin** statement. There are no await statements outside of the **parbegin** statement. Deadlock can

only occur if the program T goes into deadlock.
$D(S) = D(T) = ((D(S_1) \wedge D(S_2)) \vee (D(S_1) \wedge post(S_2)) \vee (D(S_2) \wedge post(S_1)))$
T contains two components, where the first one does not contain any **await** statement. It means, $D(S_1) = FALSE$, the formula simplifies to $D(S) = (D(S_2) \wedge post(S_1))$. T can be in deadlock only if S_2 is in deadlock and S_1 cannot help because it already terminated, it already reached its postcondition.
Now we are interested in whether the program S_2 can get in deadlock, we want to express $D(S_2)$. We apply the deadlock formula again: S_2 does not contain further parallel blocks, but it contains an **await**. S_2 is blocked when it is waiting because the precondition of its **await** holds, meaning it is enabled to be executed, but its guard is $false$.
Since the precondition of statement
await even(n) **then** $x, n := x \cdot x, n/2$ **ta**
is the invariant, $D(S_2) = (Inv \wedge \neg$even(n)$)$. But $(D(S_2) \wedge post(S_1)) = (Inv \wedge odd(n) \wedge z = x'^{n'} \wedge n = 0) = FALSE$, because 0 is an even number, $n = 0$ and $odd(n)$ cannot hold in the same time.

Proof - 2nd part of the sequence - Interference freedom conditions

We want to prove that the interference freedom holds; S_1 and S_2 do not invalidate each other's behaviour. Notice, that the two following notations aim to express the same: in case Q is true, by executing S the program terminates without error, and in the termination states the R condition is satisfied. (You are advised to use the wp notation, this is an old material not updated with that notation.)

- $\{\{Q\}\}$ S $\{\{R\}\}$
- $Q \implies wp(S, R)$

Let us write down the conditions that are needed to be proven:

1. $\{\{(Inv \wedge n \neq 0) \wedge Inv\}\}\ n, z := n-1, z \cdot x\ \{\{Inv\}\}$
   The precondition of S_2 is preserved.
2. $\{\{(Inv \wedge n \neq 0) \wedge Inv\}\}\ n, z := n-1, z \cdot x\ \{\{Inv\}\}$
   The precondition of the while statement in S_2 is preserved.
3. $\{\{(Inv \wedge n \neq 0) \wedge Inv\}\}\ n, z := n-1, z \cdot x\ \{\{Inv\}\}$
   The precondition of the await statement in S_2 is preserved.
4. $\{\{(Inv \wedge n \neq 0) \wedge (z = x'^{n'} \wedge n = 0)\}\}\ n, z := n-1, z \cdot x\ \{\{z = x'^{n'} \wedge n = 0\}\}$ The postcondition of S_2 is preserved.
5. $\{\{(Inv \wedge n \neq 0) \wedge (Inv \wedge n \neq 0) \wedge t = t_0\}\}\ n, z := n-1, z \cdot x\ \{\{t \leq t_0\}\}$
   Remember: to guarantee that the loop of S_2 terminates, its loop body has to decrease the variant function. To prove that the assignment in S_1 does not interfere with this, we have to show that $n, z := n-1, z \cdot x$ does not make the value of the variant function greater.

Similarly:

1. $\{\{Inv \wedge Inv\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{Inv\}\}$
2. $\{\{Inv \wedge Inv\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{Inv\}\}$
3. $\{\{Inv \wedge (Inv \wedge n \neq 0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{Inv \wedge n \neq 0\}\}$
4. $\{\{Inv \wedge (z = x'^{n'} \wedge n = 0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{z = x'^{n'} \wedge n = 0\}\}$
5. $\{\{Inv \wedge (I \wedge n \neq 0 \wedge t = t_0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{t \leq t_0\}\}$

Proof - Interference freedom

By eliminating the duplicates, we get the following conditions:

1. $\{\{(Inv \wedge n \neq 0) \wedge Inv\}\}$ $n, z := n - 1, z \cdot x$ $\{\{Inv\}\}$

2. $\{\{(Inv \wedge n \neq 0) \wedge (z = x'^{n'} \wedge n = 0)\}\}$ $n, z := n - 1, z \cdot x$ $\{\{z = x'^{n'} \wedge n = 0\}\}$
3. $\{\{(Inv \wedge n \neq 0) \wedge (Inv \wedge n \neq 0) \wedge t = t_0\}\}$ $n, z := n - 1, z \cdot x$ $\{\{t \leq t_0\}\}$

4. $\{\{Inv \wedge Inv\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{Inv\}\}$
5. $\{\{Inv \wedge (Inv \wedge n \neq 0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{Inv \wedge n \neq 0\}\}$
6. $\{\{Inv \wedge (z = x'^{n'} \wedge n = 0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{z = x'^{n'} \wedge n = 0\}\}$
7. $\{\{Inv \wedge (I \wedge n \neq 0 \wedge t = t_0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{t \leq t_0\}\}$

Let us prove them separately! (Please notice, here you just have to prove conditions in the form of $Q \implies wp(S, R)$, there were several examples for them during the semester, not all of them all detailed here. The aim of thi section was to familiarize you with the concept of interference freedom.)

1. $\{\{(Inv \wedge n \neq 0) \wedge Inv\}\}$ $n, z := n - 1, z \cdot x$ $\{\{Inv\}\}$
   We already proved this when the total correctness of S_1 was proved.
2. $\{\{(Inv \wedge n \neq 0) \wedge (z = x'^{n'} \wedge n = 0)\}\}$ $n, z := n - 1, z \cdot x$ $\{\{z = x'^{n'} \wedge n = 0\}\}$ The left side is FALSE as $n \neq 0$ and $n = 0$ cannot happen in the same time. This condition describes a case that never happens.
3. $\{\{(Inv \wedge n \neq 0) \wedge (Inv \wedge n \neq 0) \wedge t = t_0\}\}$ $n, z := n - 1, z \cdot x$ $\{\{t \leq t_0\}\}$
   We already proved this when the total correctness of S_1 was proved.
4. $\{\{Inv \wedge Inv\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{Inv\}\}$
5. $\{\{Inv \wedge (Inv \wedge n \neq 0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{Inv \wedge n \neq 0\}\}$
6. $\{\{Inv \wedge (z = x'^{n'} \wedge n = 0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{z = x'^{n'} \wedge n = 0\}\}$
7. $\{\{Inv \wedge (I \wedge n \neq 0 \wedge t = t_0)\}\}$ **await** $even(n)$ **then** $x, n := x \cdot x, n/2$ **ta** $\{\{t \leq t_0\}\}$
   We already proved this when the total correctness of S_2 was proved.

   The proof of not proved conditions is left to the reader.