# UE Formale Methods in Computer Science

**185.A95 FMI UE – WS 2019**

**Exercise Sheet for Block 3**
**Due December 16, 2019**

---

## *** Add YOUR NAME and STUDENT ID***

## Submission instructions and guidelines

The deadline for submitting your solutions is December 16, 2019. Please upload your solutions on TUWEL, as one zip file. The zipped archive should contain:

1. One pdf file containing your written solutions using the provided LaTeX-template (on TUWEL). This file should contain your solution to the exercises, together with explanations to the Dafny files solving your exercises.
2. The Dafny files you had to write and test for solving exercises.

Please name your zipped solutions as: `surname - studentID.zip`, where `surname` is your surname and `studentID` is your student ID. Files in the zip should be named accordingly:

- `surname-studentID.pdf` – for the pdf containing your solutions;
- `surname-studentID-exnr.dfy` – for the Dafny files containing your solutions, where `exnr` is the number of the exercise sub-problem your Dafny solution refers to.

**A total number of 15 points can be achieved by solving the exercises on this sheet.**

## Exercise Sheet Problems

**Exercise 1. (5 points)** Let $p$ be the IMP program:

$$
\begin{array}{l}
c := 0; n := 0; k := 0; m := 0; \\
\textbf{while } n < N \textbf{ do} \\
\quad c := c + k; \\
\quad k := k + m; \\
\quad m := m + 6; \\
\quad n := n + 1 \\
\textbf{od}
\end{array}
$$

(a) The program $p$ is supposed to satisfy the total correctness assertion $[N \geq 0]\ p\ [c = N * N * N]$. There are, however, two mistakes in $p$. Correct $p$ such that the total assertion holds.

(b) Write a Dafny method with input argument $N$ and return value $c$, implementing your corrected program from (a).

(c) Use Dafny to prove the total correctness assertion $[N \geq 0]\ p\ [c = N * N * N]$, where $p$ is your corrected program from (a).

(d) What inductive loop invariant and loop variant did you use in part (c)? Give a formal proof, using pen-and-paper and weakest preconditions, that your invariant and variant are strong enough to prove $[N \geq 0]\ p\ [c = N * N * N]$, where $p$ is your corrected program from (a).

**Submission guidelines:** Provide the Dafny file you used in parts (b) and (c). Write and explain your solutions to the other parts in the LaTeX-template.

**Solution**: your solution.

**Exercise 2. (5 points)** Let $p$ be the IMP program:

$$
\begin{array}{l}
r := 0; a := x; b := y; \\
\textbf{while } (a > 0 \ \vee \ b > 0) \textbf{ do} \\
\quad \textbf{if } a \geq b \textbf{ then} \\
\quad\quad r := r + 1; a := a - 1 \\
\quad \textbf{else} \\
\quad\quad r := r + 1; b := b - 1 \\
\quad \textbf{fi}
\end{array}
$$

(a) Prove that the Hoare triple $\{true\}\ p\ \{r = x + y\}$ is not valid. Hint: Provide a counterexample.

(b) Write a Dafny method with input arguments $x$ and $y$ and return value $r$, implementing the program $p$.

(c) Use Dafny to prove the total correctness assertion $[x \geq 1 \wedge y \geq 1]\ p\ [r = x + y]$

(d) Give a formal proof, using pen-and-paper and weakest (liberal) preconditions, of the partial correctness assertion $\{x \geq 1 \wedge y \geq 1\}\ p\ \{r = x + y\}$.

(e) State the weakest precondition $P$ such that $\{P\}\ p\ \{r = x + y\}$ holds. You are *not* required to prove that $P$ is the weakest precondition.

(f) Replace in $p$ the condition of the while loop, i.e., $a > 0 \ \vee \ b \geq 0$, with a different condition such that for the resulting program $p'$, the Hoare triple $\{true\}\ p'\ \{r = x + y\}$ is valid. Do *not* change any other statement or condition in $p$. You do *not* have to provide a formal proof of validity.

**Submission guidelines:** Provide the Dafny file you used in parts (b) and (c). Write and explain your solutions to the other parts in the LaTeX-template.

**Solution**: your solution.

**Exercise 3. (5 points)**
Consider the Dafny program in Listing 1 on page 3. (Note: You should copy the program from the LaTeX-template, not from the PDF; otherwise, Dafny will complain about syntax errors.)

(a) Prove the total correctness of the method `threeA`.

Hint: You may call the function `power` in preconditions, post conditions and loop invariants. This is illustrated in the `ensures` statement.

(b) What does the method `threeB` compute? Prove the total correctness of the method `threeB`. To this end, add an appropriate `ensures` statement as well as loop invariants to the code.

**Submission guidelines:** Provide a Dafny file with your solution.

**Solution**: your solution.

Listing 1: The Dafny program from Exercise 3

```
function power(b: int, e: nat): int
{
  if e = 0 then 1 else b*power(b,e-1)
}

method threeA(A: int) returns (r: int)
requires A > 0
ensures r = power(A, 3)
{
  r := 1;
  var i := 3;
  while i > 0
    invariant i ≥ 0
    invariant r = power(A, 3 - i)
  {
    i := i - 1;
    r := r * A;
  }
}

method threeB(E: int, F: int) returns (r: int)
  requires E > 0 ∧ F > 0
{
  r := 1;
  var e := E;
  var f := 0;
  while e > 0
  {
    f := 0;
    while f < F
    {
      r := r * 2;
      f := f + 1;
    }

    e := e - 1;
  }
}
```