Program Verification	SS 2021	LVA 703083+703084
Sheet 13		Deadline: June 15, 2021, 8am

- Prepare your solutions on paper.
- Marking an exercise in OLAT means that a significant part of that exercise has been treated.
- Upload your solution in OLAT as a single PDF file.

Exercise 1 *Proof Tableaux* **7 p.**

Consider the following algorithm *Copy*

```
a := x;
y := 0;
while (a != 0) {
  y := y + 1;
  a := a - 1;
}
```

1. Show partial correctness of *Copy*, i.e., develop a proof tableau for $(\!|x \geq 0|\!)\, \mathit{Copy}\, (\!|x = y|\!)$ using the while-rule. (3 points)
2. Show total correctness of *Copy*, i.e., develop a proof tableau for $(\!|x \geq 0|\!)\, \mathit{Copy}\, (\!|x = y|\!)$ using the while-total-rule. (2 points)
3. Does the partial correctness property $(\!|\mathsf{true}|\!)\, \mathit{Copy}\, (\!|x = y|\!)$ hold? Either argue why it does not hold, or prove it. (2 points)

Exercise 2 *Minimal-Sum Section* **8 p.**

Consider the algorithm *Min_Sum* on slide 6/38.

1. Is the program still correct, if one swaps the two assignment $t :=$ ... and $s :=$ ... within the while-loop? Provide a counter-example, where the modified program produces a wrong result, or briefly argue why it is still sound. (2 points)
2. Prove $(\!|n > 0|\!)\, \mathit{Min\_Sum}\, (\!|\mathit{Sp}_2|\!)$ where Sp_2 is the specification on slide 6/39. To this end, find suitable invariants and create a proof tableau using the while-rule for partial correctness. Also provide informal proofs for all implications that occur in the tableau. (6 points)

Exercise 3 *Non-Termination of Imperative Programs* **5 p.**

The Hoare-calculus can not only be used to prove termination (with the while-total-rule), but it can also be used to prove non-termination via the while-rule.

1. Given a set of inputs characterised by some formula φ, provide a Hoare-triple (for partial correctness) that encodes that program P does not terminate on these inputs. (it might be useful to have a look at slide 6/51 that was not yet discussed in the lecture, where termination is formulated as stand-alone property via Hoare-triples) (3 points)

2. Prove non-termination of the factorial program for all inputs $x < 0$ by constructing a suitable proof tableau. (2 points)

```
y := 1;
while (x != 0) {
  y := y * x;
  x := x - 1
}
```