



Tutorial for Program Verification Exercise Sheet 13

In this exercise sheet we use the Hoare proof system, extended by rules for the new statements `havoc` and `assume`. Then we move on to *control flow graphs*, a new representation for programs.

Submit your solution by uploading it as PDF in ILIAS.

Exercise 1: Havoc and Assume

2 Points

Provide a Hoare logic proof that shows that the following Boostan program P satisfies the precondition-postcondition pair $(\{x > 0\}, \{x > 0\})$.

```
havoc y;  
assume x > y;  
x := x - y;
```

Exercise 2: CFG for Conditional Statement

2 Points

In the lecture, we defined the notion of a control-flow graph of a given statement. This definition is not yet complete: We defined it for simple statements and for the sequential composition. The conditional statement (`if/else`) and the `while` statement are still missing. In this exercise, we define the control-flow graph for conditional statements:

Let st_1, st_2 be two statements. Let $G_1 = (Loc^1, \Delta^1, \ell_{init}^1, \ell_{ex}^1)$ be a control-flow graph for st_1 , and let $G_2 = (Loc^2, \Delta^2, \ell_{init}^2, \ell_{ex}^2)$ be a control-flow graph for st_2 such that Loc^1 and Loc^2 are disjoint. Define a control-flow graph for `if (expr) { st_1 } else { st_2 }`.