

8. Exercise sheet *Semantics and Verification of Software 2007*

Due to Wed., 13 June 2007, *before* the exercise course begins.

Exercise 8.1:

Consider Example 12.3 from the lecture:

```
c0  ≡  begin
      var x;
      proc P is y := x;
      begin
        var x;
        x := 2;
        call P
      end
    end
```

Choose initial variable environment ρ_0 such that with static scope semantics, for all π_0 and σ_0 :

$$\rho_0, \pi_0 \vdash \langle c_0, \sigma_0 \rangle \rightarrow \sigma_0[0/1, 1/1, 2/2]$$

Give a formal proof, using the operational semantics in Definition 12.4.

Exercise 8.2:

Consider the following modification to our *WHILE* language, where procedures have (exactly) two *call-by-value* parameters:

```
p ::= proc P(x1, x2) is c; p | ε ∈ PDec
c ::= ... | call p(a1, a2) ∈ Cmd
```

- (a) Define new *call* and *block* rules.
- (b) Evaluate $\langle c_0, \sigma_\emptyset \rangle$ for environment $\rho_\emptyset, \pi_\emptyset$ and

```
c0  ≡  begin
      var y;
      proc MAX(x1, x2) is if x1 > x2 then y := x1 else y := x2;
      call MAX(1, 2);
    end
```