Write your student ID on each sheet you hand in.

Question 1. Evaluate the following term in big-step semantics for call-by-value.

$$(\lambda\, f\, .\, (f\ f))\ (\lambda\, z\, .z)$$

[30%]

Question 2. Let M be some λ-term, and suppose we want to evaluate the following term, in which $(M\,1)$ occurs twice:

$$((M\,1)\ (M\,1))$$

When we evaluate the term in the pure λ-calculus, we can reuse the value V that we get from evaluating $(M\,1) \Downarrow V$ the first time. We do not need to compute it again for the second occurrence of $(M\,1)$. But now suppose we have the same situation in the big-step semantics with assignment. Explain whether we can still re-use the value for the first occurrence of $(M\,1)$ for the second occurrence as well. You should make your answer precise by referring to the semantics rules.

[30%]

Question 3. Consider the following expression in a programming language based on the λ-calculus:

$$(\lambda\, x\, .\, 5)\, (g\, 1)$$

Suppose we try to optimize the expression by replacing it by 5. Is it always correct to perform this simplification? In particular, explain whether it is correct in the following situations:

- in the pure λ-calculus, based on beta reduction;
- in a programming language with state (as given by the big-step semantics with assignment);
- in a programming language with control (as given by the CEK machine with `go` and `here`).

[40%]