In order to be admitted to the exam, you have to successfully submit your homework every week, except for 2 weeks. A successful submission is one where you get at least 1 point.

**Handin** Please submit this homework until Thursday, November 08, either via email to Philipp Schuster (philipp.schuster@uni-tuebingen.de) before 12:00, or on paper at the beginning of the lab.

**Groups** You can work in groups of up to 2 people. Please include the names and Matrikelnummern of all group members in your submission.

**Points** For each of the Tasks you get between 0 and 2 points for a total of 6 points. You get:
- 1 point, if your submission shows that you tried to solve the task.
- 2 points, if your submission is mostly correct.

### Task 1: Derivation trees

We define an example language by the following grammar:

\[
\langle \text{term} \rangle ::= \text{`zero'} | \text{`succ'} \langle \text{term} \rangle | \text{`false'} | \text{`true'} \\
| \text{`iszero'} \langle \text{term} \rangle | \text{`if'} \langle \text{term} \rangle \text{`then'} \langle \text{term} \rangle \text{`else'} \langle \text{term} \rangle
\]

We define an operational semantics for the language by defining the reduction relation \( \rightarrow \) as the smallest relation \( t \rightarrow t' \), closed under the following derivation rules:

\[
\begin{align*}
\text{E-Succ} & \quad t_1 \rightarrow t'_1 & \text{E-IsZeroZero} & \quad \text{iszero zero} \rightarrow \text{true} \\
\text{succ} t_1 & \rightarrow \text{succ} t'_1 & \text{E-IsZeroSucc} & \quad \text{iszero(succ } t) \rightarrow \text{false} \\
\text{E-IfTrue} & \quad \text{if true then } t_2 \text{ else } t_3 \rightarrow t_2 & \text{E-IfFalse} & \quad \text{if false then } t_2 \text{ else } t_3 \rightarrow t_3 \\
\text{E-If} & \quad t_1 \rightarrow t'_1 & & \text{if } t_1 \text{ then } t_2 \text{ else } t_3 \rightarrow \text{if } t'_1 \text{ then } t_2 \text{ else } t_3
\end{align*}
\]

Which of the rules are computation rules, which of the rules are congruence rules?
Prove that the term \( \text{succ}(\text{succ}(\text{iszero}(\text{succ zero}))) \) is not in normal form, by giving a derivation tree with root:

\[
\text{succ}(\text{succ}(\text{iszero}(\text{succ zero}))) \rightarrow \text{succ}(\text{succ false})
\]

**Task 2: Deterministic reduction**

The language and its reduction relation from Task 1 is non-deterministic which means that there is a term \( t \) that reduces in one step to two different terms. Show this, by finding \( t, t_1 \) and \( t_2 \) such that \( t \rightarrow t_1 \) as well as \( t \rightarrow t_2 \). No proof required.
Describe in two sentences an approach for making the reduction relation deterministic.

**Task 3: Induction on derivation trees**

Let the function \( \text{size} \) for the language from Task 1 be defined as:

\[
\begin{align*}
\text{size}(\text{zero}) &= 1 \\
\text{size}(\text{succ } t_1) &= \text{size}(t_1) + 1 \\
\text{size}(\text{false}) &= 1 \\
\text{size}(\text{true}) &= 1 \\
\text{size}(\text{iszero } t_1) &= \text{size}(t_1) + 1 \\
\text{size}(\text{if } t_1 \text{ then } t_2 \text{ else } t_3) &= \text{size}(t_1) + \text{size}(t_2) + \text{size}(t_3) + 1
\end{align*}
\]

Show by induction on the possible derivation trees that from \( t \rightarrow t' \) it follows that \( \text{size}(t') < \text{size}(t) \).