# Background Questionaire

Please try to answer the following questions alone and without using any external aids. If you have trouble, just skip a question instead of guessing and thinking very hard. Try not to write lengthy answers, often bullet-points are enough. This is no exam, please hand in the results *anonymously*. The results are used only to adapt the *Interactive Theorem Proving* course to the background and interests of the audience.

## 1 Functional Programming

1. Consider the following functional program on lists.

   ```
   fun SNOC x [] = [x]
     | SNOC x (y::ys) = y::(SNOC x ys)
   ```

   Please answer the following questions:

   - What is the result of `SNOC 5 [7,3,2]`?
   - Describe informally (and very briefly) what the function `SNOC` does.

2. Write a functional program `APPEND` that appends two lists. Try to use SML notation. `APPEND` should satisfy the following example behaviours
   - `APPEND [1,2,3] [4,5] = [1,2,3,4,5]`
   - `APPEND [] [1] = [1]`

3. What is *tail-recursion* and why is it important for functional programming?

4. Write a tail-recursive version of `APPEND`.

## 2 Induction Proofs

1. Prove that the following method to calculate the sum of the first $n$ natural numbers is correct (notice $0 \notin \mathbb{N}$, i. e. $\mathbb{N} = \{1, 2, 3, \ldots\}$):

   $$\forall n \in \mathbb{N}.\ \sum_{1 \leq i \leq n} i = \frac{n * (n+1)}{2}$$

2. Prove that `∀x l. LENGTH (SNOC x l) = LENGTH l + 1` holds for the function `SNOC` given above. You can use arithmetic facts and the following properties of `LENGTH`:
   - `LENGTH [] = 0`
   - `∀x xs. LENGTH (x :: xs) = LENGTH xs + 1`

3. Prove that `APPEND` is associative (hint: via induction). You can use the following properties of `APPEND`. Use the notation `l1 ++ l2` for `APPEND l1 l2`.
   - `∀l. [] ++ l = l`
   - `∀l x xs. (x :: xs) ++ l = x :: (xs ++ l)`

4. Which induction principles do you know? Which ones did you use above?

# 3 Logic

1. Explain briefly what's wrong with the following reasoning: "No cat has two tails. A cat has one more tail than no cat. Therefore, a cat has three tails."

2. It is well known that in ancient times (a) all *Spartans* were *brave* and (b) all *Athenians* were *wise*. Spartans and Athenians always fought with each other. So there was (c) no dual citizenship. Once upon a time, 3 Greek philosophers met: Diogenes, Platon and Euklid. In contrast to their famous namesakes, not much is known about them. We know however, that (d) they all came from Sparta or Athens. During their meeting, the 3 philosophers started to argue and finally insulted each other. Being philosophers they were very careful not to tell a lie, though. A few fragments of what they said have come to us through the centuries:

   (e) Euklid: "If Platon is from Sparta, then Diogenes is a coward."

   (f) Platon: "Diogenes is a coward, provided Euklid is from Sparta."

   (g) Platon: "If Diogenes is from Athens, then Euklid is a coward."

   (h) Diogenes: "If Platon is from Athens, then Euklid is a moron."

   Can you reconstruct from which town each of these philosophers came?

   a) Formalise the relevant parts of the text above in first order logic. Model *is coward* as *is not brave* and *is moron* as *is not wise*.

   b) Using the proof method of *resolution* show that Platon is from Sparta. If you don't know the resolution method, try to show it using some other method.

   c) Which town did Euklid come from?

3. Let the function *myst* for all $R$ of type $\alpha \to \alpha \to \mathit{bool}$ be given by

$$\mathit{myst}(R) = \lambda a\, b.\ \forall Q.\ \left( \begin{array}{c} \forall x.\ Q\, x\, x \quad \wedge \\ \forall x\, y.\ R\, x\, y \implies Q\, x\, y \quad \wedge \\ \forall x\, y\, z.\ (Q\, x\, y \ \wedge\ Q\, y\, z) \implies Q\, x\, z \end{array} \right) \implies Q\, a\, b$$

   a) Which type does *myst* have?

   b) What concept does the type $\alpha \to \alpha \to \mathit{bool}$ represent?

   c) Translate the formula in English using as high level concepts as possible.

   d) What concept does the function *myst* define?

# 4 General

Do you have any comments or suggestions? Is there something that you believe is relevant for selecting and prioritising topics of the Interactive Theorem Proving course. I'm happy if you don't write anything here.