Logical reasoning and programming, lab session 9

(November 14, 2022)

The goal of this lab session is to play a bit with proof assistants. We use Isabelle, which has a brief tutorial available here. In fact, Isabelle is a generic framework and we are using only Isabelle/HOL here; a particular object logic. One more thing you can give a try is Lean with an online tutorial available here (Lean 3). Note that Lean 4 is not backward compatible with Lean 3 (used in `mathlib`).

9.1 Try examples from the tutorial. In particular, you should try examples (and exercises) on the use of induction in Isabelle. Moreover, you can for example verify that the reverse of list using an accumulator produces the same result as the standard definition. Feel free to break things by stating false theorems (e.g., `rev(xs) = xs`) and try to find counter-examples for them.

9.2 Open theory `Isabelle2022/src/HOL/Metis_Examples/Sets.thy` (HTML version) and play with `sledgehammer`.

9.3 Open `Isabelle2022/src/HOL/Examples/Sqrt.thy` (HTML version) and play with it. It is possible to prove various steps using `try0`, `try`, and also `sledgehammer`. Try them!

9.4 Open `Isabelle2022/src/HOL/Isar_Examples/Group.thy` (HTML version) and play with it. You can try to use the automation available in Isabelle to prove or refute (using `quickcheck` and `nitpick`) things. You can go through examples that we discussed at the previous labs like `x * y = y * x`.

9.5 Open `Isabelle2022/src/HOL/Nitpick_Examples/Integer_Nits.thy` (HTML version) and play with it using the automation available in Isabelle.

9.6 For example, you can also check a simplified model of Unix filesystem in `Isabelle2022/src/HOL/Unix/Unix.thy` (HTML version).

9.7 An example of a well-known theorem with many dependencies, which are resource intensive to check, is the Central Limit Theorem (CLT), see `Isabelle2022/src/HOL/Probability/Central_Limit_Theorem.thy` (HTML version).

9.8 You can check a large collection of proofs for Isabelle available in the Archive of Formal Proofs.