Exercise sheet 4

Computational Metaphysics

Benzmüller, Steen, Wisniewski Summer 2016

The exercises will be discussed in the tutorial session (wednesday 2pm).

Additional natural deduction rules for this exercise sheet:

$$\dfrac{\begin{matrix}[\neg A] \\ \vdots \\ A\end{matrix}}{A}\ \mathit{classical} \qquad \dfrac{\neg A \quad A}{B}\ \mathit{notE}^a$$

[a] This rule was already mentioned before, we now can use this stronger version with the B in its conclusion as given by Isabelle.

Please solve all the following excercises using the Isabelle *system. Add all your solutions to the same* `.thy` *file, create a* `.pdf` *file and upload both using the KVV system. The* `.thy` *file (and thus the theory) should be named using the format* $\texttt{Lastname}_1\texttt{Lastname}_2\texttt{Ex04.thy}$ *as in* `MüllerMeierEx04.thy`*. If you are using a temporary account, please also state the account name somewhere in your solution.*

Exercise 1: Ontological Argument I.

During the long history of ontological arguments a great variety of different formalizations has occured. On given in propositional logic can be stated as

"If god does not exist, then it's not the case, that if I pray my prayers will be answered.
I do not pray, therefore god exists."

This can be formalized as

$$\neg G \to \neg(P \to A),\ \neg P \vdash G.$$

Please formalize a proof in Isabelle.
Now that you have proved the conjecture, are you completely convinced that God exists (or more non-theistic: that the formalization is adequate)? Please justify your answer and point out, what could be the cause of these problems.

Exercise 2: Induction proofs.

In this exercise, we investigate how induction proofs can be formalized in Isabelle. To that end, please read the Chapter 4.4. from the prog-prove manual[1]. You may use the `rule` (as done for ND proofs) and `simp` proof tactics.

(a) Go through the details of the proof of $\Sigma_{i=1}^{n} i = \frac{n(n+1)}{2}$ stated in Chapter 4.4 of prog-prove. Afterwards, close the manual and formalize a similar proof on your own. This time, we define as special function sum_n[2] given by
```
fun sum_n :: "nat ⇒ nat" where
```

[1] See `https://isabelle.in.tum.de/dist/Isabelle2016/doc/prog-prove.pdf`

[2] We do that do abstract of the special syntax sugar that Isabelle offers to write down sums more easily. In our case, we want to learn induction proofs and not get distracted by this syntactic sugar.

```
"sum_n 0 = 0" |
"sum_n (Suc n) = Suc n + sum_n n"
```
Prove the following lemma in Isabelle using the `induction` method:
```
lemma "sum_n n = n * (n + 1) div 2"
```

(b) Define a similar function `sum_n_square :: nat` $\Longrightarrow$ *nat* that implements $\Sigma_{i=1}^{n} i^2$. Prove that `sum_n_square` $n = \frac{n(n+1)(2n+1)}{6}$ holds in Isabelle. This might be tricky! A little hint: Try to get rid of the division by 6 as a first step.

Exercise 3: Riddle No. 2.

The following riddle is valid in classical logic:

"If every poor person has a rich parent,
then there is a rich person who has a rich grandparent."

This statement is formalized by

$$\forall X.\ \neg \text{rich } X \longrightarrow \text{rich (parent } X) \vdash \exists X.\ \text{rich (parent (parent } X)) \wedge \text{rich } X$$

Please first sketch a pen-and-paper (or the electronical counterpart) proof and then formalize your proof using Isabelle. Please use a proof by contradiction and/or by case distinctions in both cases. You are allowed use the *simp* rule in your solution, together with the calculus rules introduced so far.

Hint: Proof first, that there exists a rich person.