Isabelle/HOL Exercises
Logic and Sets

Propositional Logic

In this exercise, we will prove some lemmas of propositional logic with the aid of a calculus of natural deduction.

For the proofs, you may only use

- the following lemmas:
  notI: $(A \Longrightarrow False) \Longrightarrow \neg A$,
  notE: $[\![\neg A; A]\!] \Longrightarrow B$,
  conjI: $[\![A; B]\!] \Longrightarrow A \wedge B$,
  conjE: $[\![A \wedge B; [\![A; B]\!] \Longrightarrow C]\!] \Longrightarrow C$,
  disjI1: $A \Longrightarrow A \vee B$,
  disjI2: $A \Longrightarrow B \vee A$,
  disjE: $[\![A \vee B; A \Longrightarrow C; B \Longrightarrow C]\!] \Longrightarrow C$,
  impI: $(A \Longrightarrow B) \Longrightarrow A \longrightarrow B$,
  impE: $[\![A \longrightarrow B; A; B \Longrightarrow C]\!] \Longrightarrow C$,
  mp: $[\![A \longrightarrow B; A]\!] \Longrightarrow B$
  iffI: $[\![A \Longrightarrow B; B \Longrightarrow A]\!] \Longrightarrow A = B$,
  iffE: $[\![A = B; [\![A \longrightarrow B; B \longrightarrow A]\!] \Longrightarrow C]\!] \Longrightarrow C$
  classical: $(\neg A \Longrightarrow A) \Longrightarrow A$
- the proof methods *rule*, *erule* and *assumption*.

Prove:

```
lemma I: "A ⟶ A"
  apply (rule impI)
  apply assumption
done

lemma "A ∧ B ⟶ B ∧ A"
  apply (rule impI)
  apply (erule conjE)
  apply (rule conjI)
  apply assumption
  apply assumption
```

```
done

lemma "(A ∧ B) ⟶ (A ∨ B)"
  apply (rule impI)
  apply (erule conjE)
  apply (rule disjI1)
  apply assumption
done

lemma "((A ∨ B) ∨ C) ⟶ A ∨ (B ∨ C)"
  apply (rule impI)
  apply (erule disjE)
  apply (erule disjE)
  apply (rule disjI1)
  apply assumption
  apply (rule disjI2)
  apply (rule disjI1)
  apply assumption
  apply (rule disjI2)
  apply (rule disjI2)
  apply assumption
done

lemma K: "A ⟶ B ⟶ A"
  apply (rule impI)+
  apply assumption
done

lemma "(A ∨ A) = (A ∧ A)"
  apply (rule iffI)

  apply (erule disjE)
  apply (rule conjI)
  apply assumption+
  apply (rule conjI)
  apply assumption+

  apply (erule conjE)
  apply (rule disjI1)
  apply assumption
done

lemma S: "(A ⟶ B ⟶ C) ⟶ (A ⟶ B) ⟶ A ⟶ C"
```

```
  apply (rule impI)+
  apply (erule impE)
  apply assumption
  apply (erule impE)
  apply assumption
  apply (erule impE)
  apply assumption+
done

lemma "(A ⟶ B) ⟶ (B ⟶ C) ⟶ A ⟶ C"
  apply (rule impI)+
  apply (erule impE)
  apply assumption
  apply (erule impE)
  apply assumption+
done

lemma "¬ ¬ A ⟶ A"
  apply (rule impI)
  apply (rule classical)
  apply (erule notE)
  apply assumption
done

lemma "A ⟶ ¬ ¬ A"
  apply (rule impI)
  apply (rule notI)
  apply (erule notE)
  apply assumption
done

lemma "(¬ A ⟶ B) ⟶ (¬ B ⟶ A)"
  apply (rule impI)+
  apply (rule classical)
  apply (erule impE)
  apply assumption
  apply (erule notE)
  apply assumption
done

lemma "((A ⟶ B) ⟶ A) ⟶ A"
  apply (rule impI)+
  apply (rule classical)
```

```
  apply (erule impE)
  apply (rule impI)
  apply (erule notE, assumption)+
done

lemma "A ∨ ¬ A"
  apply (rule classical)
  apply (rule disjI2)
  apply (rule notI)
  apply (erule notE)
  apply (rule disjI1)
  apply assumption
done

lemma "(¬ (A ∧ B)) = (¬ A ∨ ¬ B)"
  apply (rule iffI)

  apply (rule classical)
  apply (erule notE)
  apply (rule conjI)
  apply (rule classical)
  apply (erule notE)
  apply (rule disjI1)
  apply assumption
  apply (rule classical)
  apply (erule notE)
  apply (rule disjI2)
  apply assumption

  apply (rule notI)
  apply (erule conjE)
  apply (erule disjE)
  apply (erule notE, assumption)+
done
```