MP 1 – Basic Isabelle Proofs

CS 477 – Spring 2020

Revision 1.0

Assigned February 13, 2020
Due February 20, 2020, 9:00 PM
Extension 48 hours (penalty 20% of total points possible)

1 Change Log

1.0 Initial Release.

2 Objectives and Background

The purpose of this MP is to test the student's ability to

- start up and interact with Isabelle;
- write proofs of propositional formulae using the basic proof methods.

Another purpose of MPs in general is to provide a framework to study for the exam. Several of the questions on the exam will appear similar to the MP problems.

3 Turn-In Procedure

The pdf for this assignment (`mp1.pdf`) should be found in the `assignments/mp1/` subdirectory of your git directory for this course, along with a skeleton version of the file `mp1.thy`. You should put code answering each of the problems below in the file `mp1.thy`. Your completed `mp1.thy` file should be put in the `assignments/mp1/` subdirectory of your git directory (where it was originally found) and committed as follows:

```
git pull
git add mp1.thy
git commit -m "Turning in mp1"
git push
```

Please read the *Instructions for Submitting Assignments* in

`http://courses.engr.illinois.edu/cs477/mps/index.html`

4 Propositional Logic in Isabelle/HOL

Isabelle/HOL supports Natural Deduction proofs for propostional logic with the following introduction and elimination rules:

`notI`:	$(P \Longrightarrow False) \Longrightarrow \neg P$
`notE`:	$[\![\neg P; P]\!] \Longrightarrow R$
`conjI`:	$[\![P; Q]\!] \Longrightarrow P \wedge Q$
`conjE`:	$[\![P \wedge Q; [\![P; Q]\!] \Longrightarrow R]\!] \Longrightarrow R$
`disjI1`:	$P \Longrightarrow P \vee Q$
`disjI2`:	$Q \Longrightarrow P \vee Q$
`disjE`:	$[\![P \vee Q; P \Longrightarrow R; Q \Longrightarrow R]\!] \Longrightarrow R$
`impI`:	$(P \Longrightarrow Q) \Longrightarrow P \longrightarrow Q$
`impE`:	$[\![P \longrightarrow Q; P; Q \Longrightarrow R]\!] \Longrightarrow R$
`iffI`:	$[\![P \Longrightarrow Q; Q \Longrightarrow P]\!] \Longrightarrow P = Q$
`iffE`:	$[\![P = Q; [\![P \longrightarrow Q; Q \longrightarrow P]\!] \Longrightarrow R]\!] \Longrightarrow R$

I have tried to use the same names for the variables as in Isabelle, but variable names are subject to change from one version to another, so it is best to check the actual names by using `thm` before relying upon the names.

5 Problems

Note: For each of the foramulae in each of the problems below, give a complete `apply` style proof. You must remove each occurence of `sorry` and replace it with a proof. For this assignment, the proofs must by in `apply` style (as shown in class) and are to use only the restricted proof methods that are the composition of one of `rule`, `rule_tac`, `erule`, or `erule_tac` with one of the (possibly specialized) rules given above, or `assumption`.

Note: You are free to create your own lemmas in Isabelle, and use them in proofs, so long as they have complete `apply` style proofs satisfying the above restrictions. (You may apply this extension recursively. ☺)

1. (3pts) $(A \wedge B) \longrightarrow (B \wedge A)$

2. (4pts) $(A \vee B) \longrightarrow (B \vee A)$

3. (4pts) $(A \wedge B) \longrightarrow ((\neg B) \longrightarrow (\neg A))$

4. (5pts) $(A \longrightarrow B) \longrightarrow ((\neg B) \longrightarrow (\neg A))$

5. (5pts) $((A \wedge B) \longrightarrow C) \longrightarrow (A \longrightarrow (B \longrightarrow C))$

6. (7pts) $((\neg B) \vee (\neg A)) \longrightarrow (\neg(A \wedge B))$

7. (7pts) $((\neg A) \vee (\neg B)) \longrightarrow (\neg(A \wedge B))$

6 Extra Credit

The set of rules given above are sound for the standard model of propositional logic, but they are not complete. We require at least one more rule to admit what is caled "classical" reasoning. (Reasoning done with only the rules give above is referred to as "intuitionistic" reasoning.) The one Isabelle rule we will add is the following:

$$\texttt{classical:}\ (\neg A \Longrightarrow A) \Longrightarrow A$$

In the problems below, you must follow the same rules as for the problems above except that you are also allowed to use `classical`.

8. (1 pt) $\neg\neg A \longrightarrow A$

9. (2 pts) $A \vee \neg A$

10. (2 pts) $(\neg A \longrightarrow B) \longrightarrow (\neg B \longrightarrow A)$

11. (2 pts) $((A \longrightarrow B) \longrightarrow A) \longrightarrow A$

12. (5 pts) $(\neg(A \wedge B)) = (\neg A \vee \neg B)$

13. (3 pts) $(\neg A \Longrightarrow False) \Longrightarrow A$
    In this problem you will probably want to use `rule_tac`. If, for example, you wanted to use `rule_tac` with `impI`, you would first use

    ```
    thm impI
    ```

    to see the actual variable names. In this case, you would see:

    $(\texttt{?P} \Longrightarrow \texttt{?Q}) \Longrightarrow \texttt{?P} \longrightarrow \texttt{?Q}$

    Then you could use

    ```
    apply (rule_tac P = "A ∧ B" and Q = "A" in impI)
    ```

    This will cause you to use the instance of the rule `impI` that says:

    $$(A \wedge B \Longrightarrow A) \Longrightarrow A \wedge B \longrightarrow A$$