AR Coursework Lecture

Imogen Morris

School of Informatics

18th October 2018

Information

- Demonstrator/TA: Imogen Morris s1402592@ed.ac.uk
- Lab sessions: 5.05 West Lab, Appleton Tower,
- Submission Deadline: 4pm 19th Nov
- Isabelle 2018 is installed on DICE machines: type 'Isabelle FOO.thy' in the terminal window.
- You should have a look at the recommended reading and try the exercises from the course website.

Coursework Part 1

- I. Prove some propositional and first-order proofs:
- You may use only the methods

rule	**rule_tac**	**drule**
drule_tac	**erule**	**erule_tac**
frule	**frule_tac**	**cut_tac**
assumption		

Coursework Part 1

- You may use only the rules:

conjI	**conjE**	**impI**
impE	**mp**	**iffI**
iffE	**notI**	**notE**
disjI1	**disjI2**	**disjE**
exI	**exE**	**allI**
allE	**spec**	**excluded_middle**

- You may also use as rules any lemmas that you have proven in this way.
- No automatic proof methods (**auto**, **blast** etc) !

Coursework Part 2

- Formalising Some Simple Curve geometry (3 sub-parts):
- Mechanise the Basic Definitions
- Sum of curves $c = c_1 \sqcup c_2$ (functional) is represented as **c isSumOf c1 c2** (relational).
- Functional definitions are assumed to be total.
- But the sum of two curves is not defined if they have no point in common.

Coursework Part 2

- Mechanise the Axioms
- For universal quantifiers use meta level not object level e.g. $\forall x.Px$ can be formalised as Px or $\bigwedge x.Px$. Same for implication.
- This is because otherwise you would always begin your proofs with **allE** or **impE**.

Coursework Part 2

Mechanise the basic consequences of the axioms
In the paper, Corollary

$$\forall c_1 \forall c_2 \forall c_3 \forall P[c_2 \sqsubset c_3 \wedge \mathrm{meet}(P, c_1, c_2) \wedge \mathrm{meet}(P, c_1, c_3) \implies c_1 \sqcup c_2 \sqsubset c_1 \sqcup c_3]$$

You are given

```
lemma corollary_2_6_part1:
  assumes "c2 isPartOf c3" "c isSumOf c1 c2"
          "c' isSumOf c1 c3"
  shows "c isPartOf c'"
```

Assumptions that the curves meet are no longer needed because **isSumOf** is a predicate and implies existence of a meeting point.

Searching for useful theorems

- In Parts 2 and 3 of the coursework, you are allowed to use any of the theorems in the imported theory **Main**.
- E.g. to prove Remark 2.8 you might want to find theorems about set cardinality **card**.
- You can find the theorems by searching for **card** in the query box.
- You can also search for statements using _ as a wildcard.

Coursework Part 3

- A challenge proof: Theorem 2.13.
- Pen-and-paper proof is relatively long.
- This mechanisation is non-trivial, so plan it well.
- Devise and represent the main lemmas corresponding to Steps 1 and 2.
- Credit will be given for partial mechanisation (relevant to final proof).
- Brief discussion of mechanisation vs pen-and-paper proof.

An example lemma

```
lemma assumes "c1 ≠ c2"  "c1 isPartOf c2"
  shows"∃r. r isIncidentTo c2 ∧ ¬ (r isIncidentTo c1)"
proof -
  from `c1 isPartOf c2`
  have incident_imp:"∀P. P isIncidentTo c1 ⟶ P isIncidentTo c2"
    by (subst(asm) isPartOf_def)
  from `c1 ≠ c2` axiom_c9
  have "¬(∀P. P isIncidentTo c1 = P isIncidentTo c2)"
    by (rule contrapos_nn)
  from this have "∃P. P isIncidentTo c1 ≠ P isIncidentTo c2"
    by (subst(asm) not_all)
  then obtain P where P_def: "P isIncidentTo c1 ≠ P isIncidentTo c2"
    by (rule exE)
  from incident_imp have "P isIncidentTo c2 ∨ ¬(P isIncidentTo c1)"
    by blast
  from this P_def have "P isIncidentTo c2 ∧ ¬ (P isIncidentTo c1)"
    by blast
  then show ?thesis by blast
qed
```

Overview

- Deadline: 19th Nov, 4pm
- Refer to recommended reading and self-help exercises for background and help.
- Inbuilt tactics (auto, simp etc.) can be used from Part 2 onwards.
- Use the ‘query’ box or search the imported theories (at `https://isabelle.in.tum.de/library/HOL/`) to find theorems
- Part 3 is challenging! Break it into lemmas.
- Please make use of the TA (and not just before the deadline!).