Using Lean with undergraduate mathematicians.

K. Buzzard

Jan 2019, Lean Together, Amsterdam

Using Lean in education.

This is the story of what has been going on in the maths department at Imperial College London over the last 18 months.

I'd be interested in any ideas about what to do during the next 18 months.

If anyone else who works in a mathematics department wants any help on how to introduce Lean to their undergraduates, I'd be very happy to talk.

My background

Me:

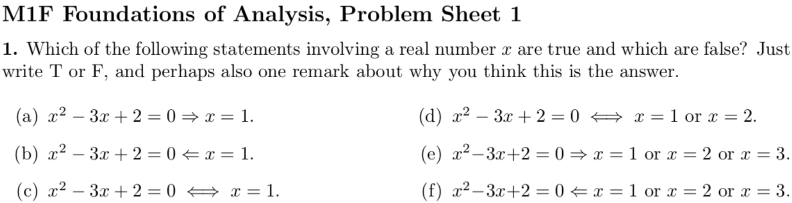
- Good maths puzzle solver,
- PhD in number theory under Richard Taylor,
- have worked for nearly 30 years in area of Langlands philosophy / Fermat's Last Theorem++ etc,
- Moved to Imperial College London 20 years ago and have never looked back.
- Both parents were teachers,
- Very interested in modern teaching methods
- Casual programmer (have contributed to Magma code base, some Python, pari-gp, Sage, 6502)
- I have taught "M1F", the introduction to proof course at Imperial, for the last three years.

How it all started

- Saw Gonthier speak in Imperial many years ago,
- Read Nov 2008 Notices of AMS with great interest,
- epsilon Haskell, then 2 weeks of Coq (whose tutorial starts in tactic mode!)
- Then this
- and I decided I wanted in.
- Read TPIL and did all the exercises. Thank you Jeremy Avigad and Soonho King and Leo!
- Asked thousands of questions on Gitter.
- Many many thanks, Mario Carneiro.

The beginnings

- Hendrik Lenstra once told me that you learn something properly only when you teach it.
- September 21st 2017 I launched blog and shortly afterwards emailed 250 incoming mathematics undergraduates telling them to come to my Thursday evening meetings.
- Spent the next week getting Lean installed on PC's in maths department computer room, setting up mailing list, page on VLE, blogging etc.
- Turned out that doing maths in Lean was *really hard*.



M1F Foundations of Analysis, Problem Sheet 1

1. Which of the following statements involving a real number x are true and which are false? Just write T or F, and perhaps also one remark about why you think this is the answer.

(a) $x^2 - 3x + 2 = 0 \Rightarrow x = 1.$

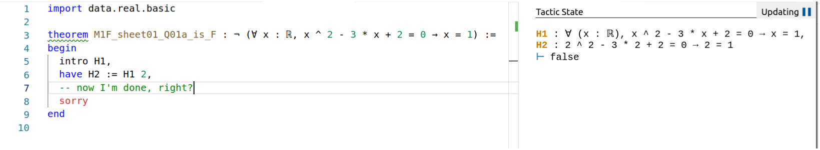
(b) $x^2 - 3x + 2 = 0 \Leftarrow x = 1.$

(c) $x^2 - 3x + 2 = 0 \iff x = 1.$

(d) $x^2 - 3x + 2 = 0 \iff x = 1$ or $x = 2.$

(e) $x^2 - 3x + 2 = 0 \Rightarrow x = 1$ or $x = 2$ or $x = 3.$

(f) $x^2 - 3x + 2 = 0 \Leftarrow x = 1$ or $x = 2$ or $x = 3.$



```
import data.real.basic

theorem M1F_sheet01_Q01a_is_F : ¬ (∀ x : ℝ, x ^ 2 - 3 * x + 2 = 0 → x = 1) :=
begin
  intro H1,
  have H2 := H1 2,
  -- now I'm done, right?
  sorry
end
```

```
H1 : ∀ (x : ℝ), x ^ 2 - 3 * x + 2 = 0 → x = 1,
H2 : 2 ^ 2 - 3 * 2 + 2 = 0 → 2 = 1
⊢ false
```

Upshot: I needed to figure out how to prove two things about real numbers:

- $2^2 - 3 \times 2 + 2 = 0$;
- $2 \neq 1$.

I asked on Gitter.

“Ooh, that’s hard” they said. And “Use `norm_num`” (which hadn’t been written). And then 50 undergraduates showed up at my Thursday evening meeting and I had to do something with them.

First successes

- 18th October 2017 a guy called Kenny Lau emails me 349 lines of Lean code, formalising the axioms of probability in Lean (which he learnt in M1S). “Please do anything with it as you please: feedback, edit, or even nothing, if this does not interest you.”
- October 31st my `xenaproject.wordpress.com` blog post about defining naturals with addition, multiplication and ordering actually teaches several of the undergraduates the basics of how all this works.
- By Christmas I had about 8 core members (Kenny and Chris Hughes included), and a few hangers-on.
- Again Mario was *absolutely indispensible* for all of this, answering question after question after question.

Problems I faced.

- Real numbers were laggy and a frequent source of time-outs.
- $(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$ was nigh-on impossible.
- I thought number theory would be easier, but students found moving between `nat` and `int` very hard. (Thanks Gabriel)
- Even some lemmas I proved in my course about functions (surjections can be split) seemed to be hard in Lean.
- A couple of Lean regulars stopped coming on Thursday evenings after the Christmas break.
- I found error messages incomprehensible.
- Lean could do magic.
- Mathlib was impenetrable.

Solutions and other positives.

- Kenny was implementing Atiyah–Macdonald's book on commutative algebra in Lean.
- Mario implemented `norm_num` and `ring` (thanks Assia for your wonderful paper!) and I subsequently wrote a blog post about what I learnt in my one and only journey into meta-land.
- Mario re-implemented the reals.
- Kenny told me "Lean does not do magic".
- Chris said to Kenny "I proved Lagrange's theorem in Lean" and Kenny said "**What?**"
- An undergraduate told me that they had learnt equivalence relations through Lean.
- I made complex numbers, and Ellen Arlt made matrices – doing things which were actually possible and useful.
- I decided to formalise schemes.
- Patrick Massot appeared and we switched to Zulip.

Interlude: Schemes

- Sounded easy enough.
- I had to finally learn typeclasses.
- Turns out that replacing an object with a canonically isomorphic object is not a triviality by any means.
- Johannes and Simon – let's get together and talk about transfer.
- Johan remarked that maybe things would be better in HoTT.
- Both Chris and Kenny made essential contributions to the project.
- More than one nugget from Reid Barton. Thanks Reid.
- I finally felt that I knew how to use Lean.
- And if something didn't work, I knew how to try and fix it, because the error messages finally all made sense.
- Now doing perfectoid spaces with Patrick and Johan, who has supported me throughout this. Thanks Patrick and Johan.

Pedagogy Transformation Grant

- My university is very interested in high-quality teaching.
- I told them I was going to introduce Lean into my Oct–Dec 2018 course and could I please have £35000.
- The money would fund:
  - 20 undergraduate “UROP” students each doing 8 week projects with me over the summer;
  - a maths education post-doc (Athina) observing my lectures and Xena sessions.
- I got funded.
- Loads of people applied for the summer jobs; I selected pretty randomly. Apart from Chris.

1st year projects.

- Each first year at Imperial has to do a project.
- Chris wrote his project on Sylow's theorems, and as part of the project he formalised all of their proofs in Lean.
- Kenny formalised the global Langlands conjectures for abelian algebraic groups over number fields in his first year project, *modulo the fact that we still do not have algebraic closure of a field.*
- Kenny got prize for best project of the year.

Summer UROP June–August 2018

- I have never had a more exhausting summer in all of my life.
- People just formalised random bits of maths.
- Some people were paid and barely showed up.
- Some people weren't being paid and showed up anyway.
- Kenny wasn't there. But Chris was.
- As were Luca Gerolla, Andreas Swerdlow, Rohan Mitta, Guillermo Barajas Ayuso, Stuart Green, Clara List, Guy Leroy, Sangwoo Jo (Jason), Ali Sever, Ned Summers, Keji Neri and Blair Shi, all of whom produced some serious Lean code much of which is available on the Xena UROP 2018 github page.
- People realised very quickly that Chris was more helpful than I was when it came to technical questions.
- For my 50th birthday I got π.

M1F 2018 – the prelude

- UROP finished on Friday, and term started three days later on the Monday.
- I had to figure out how to use Lean and mathlib in CoCalc (which William Stein had set up for me), digitise all the example sheets and set things up so it was all easy.
- My zeroth lecture was on Thursday, and it was a massive plug for Lean and Xena.
- In the evening 70 people descended on Xena and it was very busy.

Xena Oct–Dec 2018

- A whole bunch of new students.
- Was busier for longer.
- New students had far more help than those of the year before.
- Hence students learnt Lean much quicker.
- Attendance has now dropped to a core group (Abhimanyu Pallavi Sudhir, Alexandru-Andrei Bosinta, Calle Sönne, Jean Lo, Joseph Hua, Ken Lee and others) of around 6 first years, a new 2nd year (Amelia Livingston), plus the Xena regulars and some new UROP students who became regulars. Some of these people have already contributed to mathlib.
- Most nerdy thing we did was me and 6 students all editing one Lean file in CoCalc (writing dihedral groups).
- If you're ever in London on a Thursday evening during Imperial's term time, drop in.

The future of the Thursday Xena meetings?

- I realised that actually I didn't really know what to do with these undergraduates.
- But it's become a bit of a social club, so I just let them work on whatever. Some just come and never do any Lean at all. This is tolerated.
- Kenny seems to be teaching some of them Galois theory. More on this later.
- Attendance will drop off as exams begin to loom.
- But at the end of the day Chris Hughes, Kenny Lau, Abhimanyu Pallavi Sudhir, Guy Leroy, Jean Lo, Calle Sönne, Ellen Arlt and Blair Shi have all contributed to mathlib, and Chris and Kenny have both contributed substantially.
- In October 2019 I will get 70 in again on that first Thursday, and I am hoping that by then, two years after I started, this time I will actually be ready.

M1F 2018

- Course went really well.
- Did live Lean coding on several occasions.
- Unfortunately I spent a lot of time trying other innovations (mentimeter, flipped lectures – this is a class of 250 students) and still never managed to finish the example sheets.
- Athina watched with interest. Thank you thank you Athina!
- The students apparently liked the live coding sessions, but many could not make Lean do anything themselves.
- Very hard for beginners to navigate mathlib.
- Very hard for some beginners to even *install* mathlib. Mathematicians do not know `git`.
- Athina interviewed 50 students and will report back some time in the next few months.

What I learnt when teaching M1F

- I finally realised that I did not want to teach the students how to do M1F in Lean anyway.
- Too much of M1F is too hard.
- M1F is a big mess of stuff, it's lots of little topics.
- M1F is now gone. It will be replaced by a new course, which I will write.

Documentation for mathematicians.

- I had said to Patrick and others that I would write an instructional text showing the students how to do M1F questions in Lean.
- That is no longer the book I want to write.
- Maybe I want to write a book or a Wiki or whatever, which walks students through a first course in real analysis.
- Or maybe I want to write a book on some substantial algebra topic like Galois theory, with all proofs formally verified in Lean, but verified with readable code and not mangled mathlib proofs.
- Both of these ideas currently have potential; which one we do will depend on who joins in and what they want to learn.
- I envisage undergraduates contributing in a substantial way.

More ambitious ideas.

- I want to make some good instructional 5-minute videos.
- William Stein and me are going to make something better than Coursera.
- I am on the curriculum review committee whose job it is to replace M1F.
- Even if I cannot put Lean formally on the curriculum, I could formalise parts of the curriculum anyway. I see it as a useful gimmick.

Rallying cry!

Let’s teach maths undergraduates Lean!

Why? Because the staff don’t care!