Adapting Lean Tutorials to the Brazilian Case

(talk @ XXI EBL)

http://anggtwu.net/math-b.html#2025-ebl

Eduardo Ochs

Serra Negra, 2022may13

Abstract

Lean4 ([Overview]) is a proof assistant that is also a general-purpose language, and that has “metaprogramming facilities” ([Meta]) that let people extend its syntax in many ways. It is becoming very popular, and the page [Courses] has links to more than 40 courses on Lean, or using Lean.

I don’t think that these courses are adequate for Brazilians. They all start from some point like “everybody knows VSCode” and they move to non-trivial ideas very quickly. But “everybody knows the programs such and such” is not true in Brazil at all; half of the Brazilian logicians that I know only use computers in a very basic way, and here “knowledge about programs does not propagate” (see [EmacsConf2024]).

In this presentation I will show how I adapted the approach in [EmacsConf2024] – that I use to teach Maxima to undergraduate students with *very* little experience with computers – to create a short workshop on Lean ([Oficina0]) that teaches people how to install Lean4, how to navigate its manuals, how to understand its interface, and how to run and modify simplified versions of some examples from the main books on Lean.

Abstract (2)

[Overview] *Lean Documentation Overview*:
https://lean-lang.org/lean4/doc/.

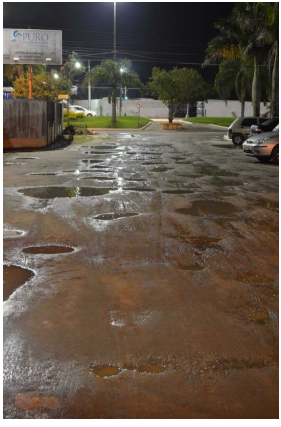
[Meta] *Metaprogramming in Lean4*:
https://leanprover-community.github.io/lean4-metaprogramming-book/

[Courses] *Courses using Lean*:
https://leanprover-community.github.io/teaching/courses.html

[EmacsConf2024] *Emacs, eev, and Maxima – Now!*:
http://anggtwu.net/emacsconf2024.html.

[Oficina0] *Oficina de Lean4 – versão 0*:
http://anggtwu.net/2024-lean4-oficina-0.html.

O PURO
(Pólo Universitário de Rio das Ostras / UFF):



Introdução
As minhas matérias principais são Cálculo 2 e 3
pra Engenharia de Produção (EP) e pra
Ciência da Computação (CC).

Em 2022.1 – primeiro semestre pós-pandemia –
alguns colegas meus abriram um processo contra mim
"porque eu estava aprovando alunos que não sabiam
o suficiente".

Agora eu começo C2 com uns "testes de nivelamento"
que me dão provas documentais que de que os alunos
estão chegando em C2 sem saberem nada da matéria do
Ensino Médio e nada de Cálculo 1.

($\rightarrow$ e eu uso o Maxima)

O PURO é um lixo e o meu departamento mais ainda

Agora os primeiros links em destaque na minha página são estes:

2025: O PURO é um lixo (links)
2025: Derive como um macaco
2025: Você atropelou a Selana
2025: Caraca, até o Walter???

O “derive como um macaco” é sobre reclamações dos alunos.

Recomendo muito!!!

Teste de nivelamento (Cálculo 2)

Por favor escrevam nesta folha...

- seu nome (legível),
- com quem você fez C1 (e GA e Prog1), em que semestre foi, e que livros você usou – e se você fez várias vezes fale sobre cada vez,
- a questão e tudo que você conseguir fazer pra resolver ela. A questão é:

$$\frac{d}{dx} f(\operatorname{sen}(x^4) + \ln x) \;=\; ?$$

$$\frac{d}{dx}\, f(\operatorname{sen}(x^4) + \ln x) \;=\; ?$$

$$\begin{array}{ccc}
\begin{pmatrix} \frac{d}{dx} f(g(x)) = \\ f'(g(x))g'(x)) \end{pmatrix} & \to & \begin{pmatrix} \frac{d}{dx} f(42x) = \\ f'(42x) \cdot 42 \end{pmatrix} \\
\downarrow & \searrow & \downarrow \\
\begin{pmatrix} \frac{d}{dx} \sin(g(x)) = \\ \cos(g(x))g'(x) \end{pmatrix} & \to & \begin{pmatrix} \frac{d}{dx} \sin(42x) = \\ \cos(42x) \cdot 42 \end{pmatrix}
\end{array}$$

Os passos intermediários são óbvios
pra gente – mas pros alunos não!!!

Outro teste de nivelamento

Lembre que:

$$\begin{aligned} f'(a) &= \lim_{\varepsilon \to 0} \frac{f(a+\varepsilon) - f(a)}{\varepsilon} \\ \left(\frac{d}{dx} f(x) \right) \Big|_{x=a} &= \lim_{\varepsilon \to 0} \frac{f(a+\varepsilon) - f(a)}{\varepsilon} \end{aligned}$$

Cada limite abaixo representa a derivada de certa função f em certo número a. Diga o que são f e a em cada caso.

a) $\displaystyle\lim_{h \to 0} \frac{(1+h)^{10} - 1}{h}$

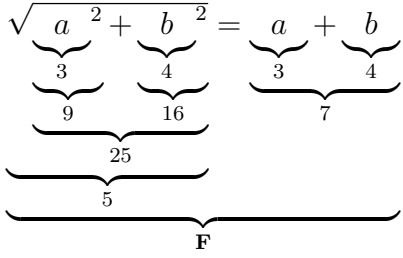
b) $\displaystyle\lim_{h \to 0} \frac{\sqrt[4]{16+h} - 2}{h}$

Aipim

Seja:

$$[\text{Aipim}] \;=\; \left(\sqrt{a^2+b^2} = a+b\right)$$

Todo mundo sabe que:



...e mesmo assim os alunos de C2 continuam usando MONTES de aipins nas suas contas, e eles não entendem o que tá errado...

Veja o “Derive como um macaco”.

Uma desculpa pra aprender Lean

Um dos objetivos do curso de C2 é a gente aprender a lidar com algumas contas muito grandes e muito complicadas...

A gente quer aprender a fazer contas fáceis de justificar e de revisar...

Mas justificar "no sentido Antônio" não serve!!!

O que vai dar pontos na prova é vocês saberem justificar os passos de vocês "no sentido calc sem rw"...

```
variable (a b c d e : Nat)
variable (h1 : a = b)
variable (h2 : b = c + 1)
variable (h3 : c = d)
variable (h4 : e = 1 + d)

include h1 h2 h3 h4 in
theorem T2 :
  a = e     := calc
  a = b     := by rw [h1]
  _ = c + 1 := by rw [h2]
  _ = d + 1 := by rw [h3]
  _ = 1 + d := by rw [Nat.add_comm]
  _ = e     := by rw [h4]
```

```
variable (a b c d e : Nat)
variable (h1 : a = b)
variable (h2 : b = c + 1)
variable (h3 : c = d)
variable (h4 : e = 1 + d)

include h1 h2 h3 h4 in
theorem T1 :
  a = e     := calc
  a = b     := h1
  _ = c + 1 := h2
  _ = d + 1 := congrArg Nat.succ h3
  _ = 1 + d := Nat.add_comm d 1
  _ = e     := Eq.symm h4
```

Manga

O ‘=’ é uma manga.

Outra manga:

$$\begin{aligned} 2(y+z) &\Rightarrow 2 \cdot (y+z) \\ f(y+z) &\Rightarrow f \ \boxed{\text{ap}} \ (y+z) \\ (a+b)[a := 42] &\Rightarrow (a+b) \ \boxed{\text{s}} \ [a := 42] \end{aligned}$$

$$(\underbrace{a+b=b+a}_{\substack{\text{expressão}\\ \text{original;}\\ \text{caso geral;}\\ \text{“antes”}}}) \quad \underbrace{[a := 42]}_{\substack{\text{substituição:}\\ a \ \boxed{\text{vira}} \ 42}} \quad = \quad (\ \underbrace{42+b=b+42}_{\substack{\text{expressão nova;}\\ \text{caso particular;}\\ \text{“depois”}}} \ )$$

“Emacs, eev, and Maxima – Now!”

http://anggtwu.net/emacsconf2024.html

Esse foi o título da minha apresentação na EmacsConf 2024...
O “Now!” queria dizer:

> “...in less than one hour and even for people
> who have never seen a terminal in their lives.”

Outra idéia importante da apresentação: “plonk”.

> I prefer this simpler definition here...
> for me “plonk” is the sound
> that a person makes when he, or she, or they
> hits the bottom of my list of priorities.

Eu aprendi a plonkar alunos que não conseguem perguntar
e ajustei o meu material pra ser fácil perguntar.

Outros cursos de Lean

Durante a pandemia eu fiz um curso de Lean –
o do Francesco Noseda, da UFRJ.

Nesse curso Lean deixou de ser algo impossível pra mim –
mas eu aprendi pouco, eu passei meses quebrando a cara
com coisas que deveriam ser óbvias pro público alvo
do curso mas pra mim eram dificílimas...

Aqui tem um monte de outros cursos de Lean:
https://leanprover-community.github.io/teaching/courses.html

Como fazer algo menos frustrante que esses cursos
e ficar amigo das pessoas certas – que interagem mais?

WSL → Debian → Emacs → eev → Maxima
→ executable notes → Lean

6. Learn the basics of Emacs and eev

The best way to learn the basics is to start by copying this
section to your ~/TODO file. This is explained in these pages:
  http://anggtwu.net/2024-first-executable-notes.html
  http://anggtwu.net/2024-restructuring.html

The "basics" are these sections of the main tutorial,

  (find-eev-quick-intro "2. Evaluating Lisp")
  (find-eev-quick-intro "3. Elisp hyperlinks")
  (find-eev-quick-intro "3.1. Non-elisp hyperlinks")

the two main keys of eev, that are `M-e' and `M-j',
the hints in the header of `M-j',
copying and pasting, that is explained here:

  Menu bar -> Edit -> Cut (C-w)
  Menu bar -> Edit -> Copy (M-w)
  Menu bar -> Edit -> Paste (C-y)
  (find-eev-quick-intro "5.2. Cutting and pasting")
  (find-emacs-keys-intro "3. Cutting & pasting")

this method for saving elisp hyperlinks to intros and info pages,

  (find-kl-here-intro "2. Try it!")
  (find-kl-here-intro "3. Info")
  (find-eev-quick-intro "5. Links to Emacs documentation")
  (find-eev-quick-intro "5.1. Navigating the Emacs manuals")

Plonkando o povo do “Não tem muito o que aprender”

[Celso:] Se a ideia é criar entusiasmo acerca da linguagem, eu acho que o melhor seria introduzir eles usando um editor mais comum, tipo o VSCode. Querendo ou não, familiaridade é muito importante quando nós estamos aprendendo algo, e introduzir a uma linguagem e a um editor ao mesmo tempo talvez seja muita coisa desconhecida ao mesmo tempo.

[Eu:] ...mas nenhum desses alunos [da CC e da EP do PURO] tem interesses em comum com os outros, e eles nao tem uma cultura de aprender juntos, de ensinarem coisas uns pros outros, e de disponibilizarem as coisas que eles fizeram em blog posts ou no github.
Entao acho que nao vale a pena eu aprender VS Code pra tentar interagir com esses alunos.
Alias, melhor: nao vale a pena eu aprender VS Code SOZINHO pra tentar interagir com esses alunos. Mas se algum dia eu ficar amigo de alguem que saiba usar VS Code e que tope me ensinar e’ outra historia.

[Thiago:] Não quis dizer que lean é fácil, quis dizer que usar o lean no vscode é fácil
Não tem muito o que aprender

[Eu:] Mas uma das minhas segundas intenções com essa oficina é fazer com que pessoas que usam VSCode ou neovim façam oficinas explicando como usar o Lean neles

Lean como um Haskell melhorado

O que é o Lean?
Ele pode ser usado como proof assistant,
Ele pode ser usado como uma linguagem “normal”,
Ele é um Haskell melhorado e com menos mangas,
Ele tem um fórum muito bom no Zulip Chat,
A sintaxe dele é extensível (“metaprogramming”),
Ele pode falar com programas externos...

Exemplo:
“Tactics & Keyframes: Visualizing Lean 4 Proofs in Blender”
(David Renshaw, 2024)

http://www.youtube.com/watch?v=KuxFWwwlEtc

O que a gente consegue aprender de Lean em poucas horas?

Importante: "Aprender a fazer (com ajuda)"
é diferente de "assistir um vídeo e dizer 'ah, entendi!' "

Resposta curta: a gente consegue aprender 1) a acessar
a documentação com elisp hyperlinks, 2) as teclas principais,
3) a rodar snippets, 4) a entender uns programas muito básicos, e
5) traduzir pra Lean demonstrações simples em Dedução Natural...

Ta-daaaa:
http://anggtwu.net/2024-lean4-oficina-0.html
Essa página tem as legendas de um vídeo
e links pro vídeo e pra outras coisas!

```
a.lean
```

```
variable (P Q R : Prop)

theorem tc1 : P ∧ Q ↔ Q ∧ P :=
  let a : P ∧ Q → Q ∧ P :=
    fun b : P ∧ Q =>
      let c : P := b.1
      let d : Q := b.2
      let e : Q ∧ P := ⟨d,c⟩
      e
  let f : Q ∧ P → P ∧ Q :=
    fun g : Q ∧ P =>
      let h : Q := g.1
      let i : P := g.2
      let j : P ∧ Q := ⟨i,h⟩
      j
  ⟨a,f⟩

#print  tc1
#reduce tc1

-- [b:P∧Q]^1  [b:P∧Q]^1   [g:Q∧P]^2  [g:Q∧P]^2    tc1 := ⟨a,f⟩  where
-- ---------  ---------   ---------  ---------      a := λb:P∧Q.e  where
--    c:P        d:Q         i:P        h:Q           c := b.1
--    ---------------        ---------------          d := b.2
--          e:Q∧P                 j:P∧Q               e := ⟨d,c⟩
--        ----------1           ----------2           f := λg:Q∧P.j  where
--        a:P∧Q→Q∧P             f:Q∧P→P∧Q               h := g.1
--        ---------------------------                   i := g.2
--                  tc1:P∧Q↔Q∧P                       j := ⟨i,h⟩
```

```
U:---  a.lean          All   L1     (Lean 4 company LSP[lean4-lsp:163612] FlyC:0 eev ElDoc) 2
```

Obrigado –
e desculpem qualquer coisa! =)