Proving theorems in Lean

Matej Penciak

Northeastern University

February 18, 2022

Slides available at:

`https://github.com/mpenciak/Lean-Seminar-Sp2022`

Plan for today, and glimpses at the future

- Learning to parse a mathlib `.lean` file
- A review of the tactics from the Natural Number Game

Comments, imports, and namespaces

```
-- single line
/-
    comment
-/

import lean.module

namespace name_space

open other_name_space

end name_space
```

Definitions, lemmas, theorems, and their proofs

```
def definition_name (assumptions : their_types)
    : type_of_output := sorry

lemma lemma_name {implicit : assumptions}
    : lemma_type :=
begin
    sorry
end

theorem theorem_name (
    same : stuff
    )
    :
    usually_Prop
    :=
    by sorry
```

sections, variables, examples, constants, #check

```
section section_name
variables (var1 : type)
end

example (no : name) : any_type : =
begin
    sorry
end

constants (G : Type) (a : G) (b : G)
variable [group G]
#check a * b
```

Attributes, lemma names, calc mode, term proofs, ...

```
variables (a b c : nat)

@[simp, to_additive]
lemma mul_right_eq_self : a * b = a ↔ b = 1 :=
calc a * b = a ↔ a * b = a * 1 : by rw mul_one
...            ↔ b = 1         : mul_left_cancel_iff
```

Structures, classes, inductive data types, instances

```
structure structure_name :=
    (constructor1 : constructor_type)
    (constructor2 : constructor_type)
    (constructor3 : constructor_type)

class class_name (any : hypotheses)
      extends other_class :=
            (constructor1 : constructor_type)

inductive inductive_name : its_type
| cons1 : type1
| cons1 : type2

instance : class_name := sorry
```

Lots of other stuff!

```
universes univ1 univ2

variables (type_depending_on_a_univ : Type univ1)

noncomputable theory

section

local attribute [simp] add_assoc

end
```

Available resources

Lots of resources with links in the demo3.lean file!

- Leanprover Community website: `https://leanprover-community.github.io/index.html`
- "Tutorial" (or Real Number Game) `https://github.com/leanprover-community/tutorials`
- "Complex Number Game" `https://github.com/ImperialCollegeLondon/complex-number-game`
- Lean for the Curious Mathematician 2020 `https://leanprover-community.github.io/lftcm2020`

More resources

- Lean Forward `https://lean-forward.github.io/`
- Lean Forward textbook, *Hithiker's Guide to Logical Verification* `https://github.com/blanchette/logical_verification_2021/blob/main/hitchhikers_guide.pdf`
- *Theoremm Proving in Lean* `https://leanprover.github.io/theorem_proving_in_lean`
- *Mathematics in Lean* `https://leanprover-community.github.io/mathematics_in_lean/basics.html`

Documentation

- Lean Reference Manual
  `https://leanprover.github.io/reference/`
- mathlib Documentation `https://leanprover-community.github.io/mathlib_docs/`