Topology Filters - Notes

Release 0.1

Carlos Caralps

Jul 16, 2021

CONTENTS

CHAPTER

ONE

INTRODUCTION

1.1 What is Lean?

Lean is an open source proof-checker and a proof-assistant. One can *explain* mathematical proofs to it and it can check their correctness. It also simplifies the proof writing process by providing `goals` and `tactics`.

Lean is built on top of a formal system called type theory. In type theory, the basic notions are "terms" and "types" — compare to "elements" and "sets" in set theory. Every term has a type, and types are just a special kind of term. Terms can be interpreted as mathematical objects, functions, propositions, or proofs. The only two things Lean can do is *create* terms and *check* their types. By iterating these two operations, we can teach Lean to verify complex mathematical proofs.

```
def x := 2 + 2                                    -- a natural number
def f (x : ℕ) := x + 3                            -- a function
def easy_theorem_statement := 2 + 2 = 4           -- a proposition
def fermats_last_theorem_statement                -- another proposition
  :=
  ∀ n : ℕ,
  n > 2
  →
  ¬ (∃ x y z : ℕ, (x^n + y^n = z^n) ∧ (x ≠ 0) ∧ (y ≠ 0) ∧ (z ≠ 0))

theorem
easy_proof : easy_theorem_statement               -- proof of easy_theorem
:=
begin
  exact rfl,
end

theorem
my_hard_proof : fermats_last_theorem_statement     -- cheating!
:=
begin
  sorry,
end

#check x
#check f
#check easy_theorem_statement
#check fermats_last_theorem_statement
#check easy_proof
#check hard_proof
```

1.2 How to use these notes

Every once in a while, you will see a code snippet like this:

```
#eval "Hello, World!"
```

Clicking on the `try it!` button in the upper right corner will open a copy in a window so that you can edit it, and Lean provides feedback in the `Lean Infoview` window. We use this feature to provide exercises inline in the notes. We recommend attempting each exercise as you go along.

These notes are based a 5-day Lean crash course at Mathcamp 2020. We have adapted them to BIYSC 2021.

These notes provide a sneak-peek into the world of theorem proving in Lean and are by no means comprehensive. It is recommended that you simultaneously attempt the Natural Number Game. It is a fun (and highly addictive!) game that proves same basic properties of natural numbers in Lean.

1.3 Acknowledgments.

These notes are based on work of Apurva Nakade and Jalex Stark. Large chunks of these notes are taken directly from <https://apurvanakade.github.io/courses/lean_at_MC2020/>`__.

1.4 Useful Links.

1. Formalizing 100 theorems
2. Formalizing 100 theorems in Lean
3. **Articles, videos, blog posts, etc.**
    1. The Xena Project
    2. The Mechanization of Mathematics
    3. The Future of Mathematics
4. Lean Zulip chat group