Fulcrum Package Usage Example

SJTU AI4Math Team

1 Knowledge Entries

1.1 Color Block Representation

Axiom 1.1 An Axiom
This is a very important axiom.
In principle, there should be very few axioms.

Definition 1.1.1 A Definition
This is the content of the definition.

Property 1.1.1.1 A property of the above definition
This is a property of the above definition.
Essentially, properties are also theorems. Generally, property entries follow the definition entry closely, describing properties that can be naturally deduced from the definition, while more important main theorems are represented by the theorem entry.

Example 1.1.1 An Example
This is an example serving the above definition.
Examples are usually added to help readers understand the content of definitions or theorems, and theoretically, removing them does not affect the construction of the main theoretical framework.

Lemma 1.1.2 A lemma serving the theorem below
This is a lemma prepared for the theorem below.
In formalist practice, we generally do not encourage referencing content that appears after the current entry. But intuitively, this lemma is prepared for the theorem below, so we reference it here.

Theorem 1.1.3 A Theorem
This is a very important theorem.

Proof:
This is a proof of the above theorem.

Corollary 1.1.4 A corollary of the above theorem
This is a corollary of the above theorem, but it is not independent or important enough to be written as a separate theorem.

Counter Example 1.1.1 A Counterexample
This is a counterexample serving the above theorem.
Counterexamples are usually added to help readers understand why certain constraints must be added to a definition or theorem, and theoretically, removing them does not affect the construction of the main theoretical framework.

2 Lean Language Support

2.1 Code Blocks

Definition 2.1.1 Definition of Cat
Fulcrum is a cat!

```
    def Fulcrum : cat := some random cat
```

Use #tm{...} to color the declared definition/theorem name.

Theorem 2.1.2 Cat Cute Theorem
All cats are very cute!

```
    theorem th_name : ∀ (x : cat), x is very cute! := by sorry
```