

All Lean code below assumes `import Mathlib`.

The following questions are *samples* of the sort of problem phrasing I may use on the midterm exam. There are more problems here than I would give on the actual exam. The exam will probably also include several short-to-medium length proofs in Lean, based on the techniques shown in class and used on homework assignments.

I may continue to add to this list in the next several days.

1. Translating statements into Lean

Convert each mathematical statement from English into a Lean statement. (You do not need to prove it.)

- (a) A product of two integers, each of which is equal to a sum of two integer squares, is again a sum of two integer squares.
- (b) Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be a function. For every real number $\varepsilon > 0$, there exists a real number $\delta > 0$ such that $|f(x) - f(y)| < \varepsilon$ whenever $|x - y| < \delta$.
Note: in Lean you can write absolute values using the `|` symbol, e.g. `|x + 1|`.
- (c) For any odd integer n , $n^2 - 1$ is divisible by 4.
- (d) The product of any two real numbers is at most the square of their average.
- (e) There is no nonzero integer b for which $2b^2$ is a perfect square.
- (f) Every natural number $n > 1$ that is not prime has a divisor m with $1 < m < n$.
- (g) There is no largest integer.

2. Tactic states

In each problem, give the tactic state (that is, what is shown in the infoview) when the cursor is on the indicated line.

(a)

```
example : ∀ x : ℝ, x^2 ≤ 2*x → ¬ x > 2 := by
  intro x hx hxpos
  -- What is the tactic state here?
```

(b) Give the tactic state at each of the lines (A) and (B).

```
example (n : ℤ) : Odd n ↔ 4 ∣ n^2 - 1 := by
  constructor
  · intro ⟨k, hk⟩
    -- (A)
  · intro hn
    rw [← Int.not_even_iff_odd]
    intro h_even
    -- (B)
```

(c)

```
example :  $\forall y : \mathbb{R}, (\forall x : \mathbb{R}, x^2 - 4x > y) \rightarrow y < -4 :=$  by
  intro y hx
  specialize hx 2
  -- What is the tactic state here?
```

(d)

```
example (n :  $\mathbb{Z}$ ) :  $\neg \text{Even } (3n) \rightarrow \neg \text{Even } n :=$  by
  intro h1
  contrapose! h1 with h2
  -- What is the tactic state here?
```

(e) The proof below has two `sorry`s. Give the tactic state at each of them.

```
example (x :  $\mathbb{R}$ ) (h :  $x^2 - 1 = 0$ ) :  $x = 1 \vee x = -1 :=$  by
  by_cases hx :  $x \geq 0$ 
  · left
    sorry
  · right
    sorry
```

(f)

```
example :  $\forall x : \mathbb{R}, x^2 + 9 \geq 6x :=$  by
  intro x
  -- What is the tactic state here?
```

(g)

```
example :  $\exists x : \mathbb{R}, x^2 + 1 < 3x :=$  by
  use 1
  -- What is the tactic state here?
```

(h)

```
example (n :  $\mathbb{N}$ ) :  $3^n \geq 2n + 1 :=$  by
  induction' n with n ih
  · norm_num
  · -- What is the tactic state here?
```

(i)

```
example (m n :  $\mathbb{Z}$ ) (h :  $m^2 = 2 * n^2$ ) :  $n = 0 :=$  by
  wlog hn :  $n \geq 0$ 
  · -- What is the tactic state here?
```

3. Defining functions

- (a) Let $S(n) = \frac{1}{1 \cdot 2} + \frac{1}{2 \cdot 3} + \cdots + \frac{1}{n(n+1)}$, so that $S(0) = 0$. Write a Lean definition of S as a function $S : \mathbb{N} \rightarrow \mathbb{Q}$.

- (b) Prove in Lean that $S(n) = 1 - \frac{1}{n+1}$ for all n :

```
lemma S_formula (n : ℕ) : S n = 1 - 1 / (n + 1) := by
```

- (c) Define a sequence by $a_0 = 0$, $a_1 = 3$, and $a_{n+2} = 7a_{n+1} - 10a_n$. Give a Lean definition of this sequence, as a function $a : \mathbb{N} \rightarrow \mathbb{Z}$.

- (d) Prove in Lean that $a_n = 5^n - 2^n$ for all n :

```
lemma a_formula (n : ℕ) : a n = 5^n - 2^n := by
```

4. One-line tasks

Each of these can be done with a single line of Lean.

- (a) Suppose the tactic state contains the hypothesis

```
h : ∃ u v : ℤ, u ^ 2 + v ^ 2 = 29
```

How do you get the following into the tactic state?

```
u v : ℤ
huv : u ^ 2 + v ^ 2 = 29
```

- (b) Suppose instead that the goal is $\vdash \exists u v : \mathbb{Z}, u^2 + v^2 = 29$. How do you change the goal to $\vdash 5^2 + 2^2 = 29$?
- (c) Suppose the tactic state contains $h : \forall x : \mathbb{R}, x^2 - 4 * x + 3 \geq y$. How do you turn this into $h : 2^2 - 4 * 2 + 3 \geq y$?
- (d) Suppose instead that the goal is $\vdash \exists x : \mathbb{R}, x^2 - 4 * x + 3 \geq y$. How do you change the goal to $\vdash 2^2 - 4 * 2 + 3 \geq y$?
- (e) Suppose the tactic state contains $h : x = 2 \vee x = -2$. How do you split the proof into two cases, one with $hx : x = 2$ and one with $hx : x = -2$?
- (f) Suppose the goal is $\vdash \neg \forall x : \mathbb{R}, x^2 > x$. How do you change the goal to $\vdash \exists x, x^2 \leq x$?
- (g) Suppose the tactic state contains $h : \neg \text{Even } n$, where $n : \mathbb{Z}$. How do you change this to $h : \text{Odd } n$?
- (h) Suppose the goal is $\vdash n = 0$, where $n : \mathbb{Z}$. How do you begin a proof by contradiction, so that the goal becomes $\vdash \text{False}$ and the tactic state contains $hn : n \neq 0$?

5. Short proofs

Write a Lean proof of each lemma.

- (a) Complete the following proof by filling in each blank. Use the `calc` block as given.

```
example {a b : ℝ} (h1 : a - b = 4) (h2 : a * b = 1) : (a + b)^2 = 20 := by
  calc
    (a + b)^2 = (a - b)^2 + 4 * (a * b) := by ____
    _ = 4^2 + 4 * 1 := by ____
    _ = 20 := by ____
```

- (b) Complete the following proof by filling in each blank. Use the `calc` block as given.

```
example {x : ℝ} (hx : x ≥ 3) : x^2 - 2*x ≥ 3 := by
  calc
    x^2 - 2*x = ____ := by ____
    _ ≥ ____ := by rel [hx]
    _ = 3 := by norm_num
```

- (c) You may use the library theorem `mul_eq_zero : a * b = 0 ↔ a = 0 ∨ b = 0`.

```
example {x : ℝ} (h : x^2 = 4*x) : x = 0 ∨ x = 4 := by
```

- (d)

```
example {n : ℤ} (hn : Odd n) : Even (n^2 + 3) := by
```

- (e)

```
example {a b : ℤ} (ha : 3 ∣ a) (hb : 3 ∣ b) : 3 ∣ a^2 + 2*b := by
```

- (f)

```
example : ¬ ∀ x : ℝ, x^2 ≥ 2*x := by
```

- (g) Complete the following proof by induction by filling in each blank.

```
example (n : ℕ) : 4^n ≥ 3*n + 1 := by
  induction' n with n ih
  · ____
  · calc
      4^(n+1) = 4 * 4^n := by ____
      _ ≥ 4 * (3*n + 1) := by ____
      _ ≥ 3*(n+1) + 1 := by ____
```