

Waterproof

Educational Software to Teach Mathematical Proof Writing

Dick Arends

22-01-2026

Radboud University

- Educational Software
- Developed in Eindhoven
- Used in Analysis 1

Lemma exercise_1 : 4 is the supremum of $[0, 4)$.

Proof.

It suffices to show that

4 is an _upper bound_ for $[0, 4)$ \wedge

$\forall \epsilon > 0, \exists a \in [0, 4), 4 - \epsilon < a.$

We show both statements.

- We need to show that $\forall c \in [0, 4), c \leq 4.$

Take $c \in [0, 4).$

We conclude that $c \leq 4.$

- We need to show that

$\forall \epsilon > 0, \exists a \in [0, 4), 4 - \epsilon < a.$

Take $\epsilon > 0.$

Either $\epsilon < 2$ or $\epsilon \geq 2.$

+ Case $\epsilon < 2.$

Choose $a := 4 - \epsilon / 2.$

{ Indeed, $a \in [0, 4).$ }

We conclude that $\& 4 - \epsilon < 4 - \epsilon / 2 = a.$

+ Case $\epsilon \geq 2.$

Choose $a := 3.$

{ Indeed, $a \in [0, 4).$ }

We conclude that $\& 4 - \epsilon < 3 = a.$

Who am I?

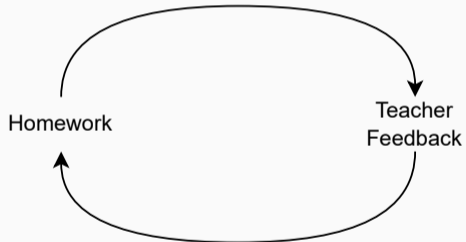
- Applied Mathematics and Computer Science in Eindhoven
- Software Engineering Project for Waterproof
- Bachelor End Project with Jim Portegies
- I have been working on Waterproof for almost 3 years
- Master Mathematical Foundations of Computer Science in Nijmegen
- Student assistant position and core member of the Waterproof team

Good entry points to learn more about Waterproof include:

- Nieuw Archief voor Wiskunde article from December 2025
- Website at `impermeable.github.io`
- PhD thesis of Jelle Wemmenhove

Students struggle with learning to write mathematical proofs.

Students struggle with learning to write mathematical proofs.



Some parts of a proof are very mechanic:

$$\forall \varepsilon \in \mathbb{R}, \exists \delta \in \mathbb{R}, \dots$$

Some parts of a proof are very mechanic:

$$\forall \varepsilon \in \mathbb{R}, \exists \delta \in \mathbb{R}, \dots$$

Proof.

Take $\varepsilon \in \mathbb{R}$.

Choose $\delta = \dots$

□

Some parts of a proof are very mechanic:

$$\forall \varepsilon \in \mathbb{R}, \exists \delta \in \mathbb{R}, \dots$$

Ideally: Student forgets to introduce the ε ? Provide immediate and clear feedback!

Solution?

Compilers give immediate feedback on the correctness of code

```
3  
4 let product = x * y;
```


⊗ example.ts 1 of 2 problems




Cannot find name 'x'. ts(2304)

Solution?

Compilers give immediate feedback on the correctness of code

```
3   
4 let product = "a string" * y;
```

 example.ts 1 of 1 problem

↓ ↑ ×

The left-hand side of an arithmetic operation must be of type 'any', 'number', 'bigint' or an enum type. ts(

Solution?

Compilers give immediate feedback on the correctness of code

```
3    
4  let product = "a string" * y;
```

 example.ts 1 of 1 problem

↓ ↑ ×

The left-hand side of an arithmetic operation must be of type 'any', 'number', 'bigint' or an enum type. ts(

Students pick this up relatively quickly

- Mechanically verifies correctness of proof steps
- Provide immediate feedback

- Mechanically verifies correctness of proof steps
- Provide immediate feedback
- Uses specialized notation
- Skills do not transfer to handwritten style proofs

- Mechanically verifies correctness of proof steps :)
- Provide immediate feedback :)
- Uses specialized notation :(
- Skills do not transfer to handwritten style proofs :(

Goal: Use Rocq to teach mathematics students to write proofs by hand

Rocq

Goal forall x : nat, x + 1 = 1 + x.

Proof.

intro x.

apply Nat.add_comm.

Qed.

Rocq

Goal forall $x : \text{nat}, x + 1 = 1 + x.$

Proof.

intro $x.$

apply `Nat.add_comm.`

Qed.

Waterproof

Goal $\forall x \in \mathbb{N}, x + 1 = 1 + x.$

Proof.

Take $x \in \mathbb{N}.$

We conclude that $x + 1 = 1 + x.$

Qed.

```
Goal forall n : nat, n > 2  
-> (0 * n = 0  $\wedge$  n + 1 > n).
```

Proof.

```
intro n.
```

```
intro H.
```

```
split.
```

```
- apply Nat.mul_0_l.
```

```
- rewrite Nat.add_1_r.
```

```
constructor.
```

Qed.

Challenges

Goal forall n : nat, n > 2
-> (0 * n = 0 \wedge n + 1 > n).

Proof.

intro n.

intro H.

split.

- **apply** Nat.mul_0_l.

- **rewrite** Nat.add_1_r.

constructor.

Qed.

n : nat

H : n > 2

=====

0 * n = 0

Goal forall $n : \text{nat}, n > 2$
-> $(0 * n = 0 \wedge n + 1 > n)$.

Proof.

intro n.

intro H.

split.

- **apply** Nat.mul_0_l.

- **rewrite** Nat.add_1_r.

constructor.

Qed.

Goal $\forall n \in \mathbb{N}, n > 2$

-> $(0 * n = 0 \wedge n + 1 > n)$.

Proof.

Take $n \in \mathbb{N}$.

Assume that $n > 2$.

We show both statements.

- We conclude that $0 * n = 0$.

- We need to show that $n + 1 > n$.

We conclude that $n + 1 > n$.

Qed.

Challenges

Goal $\forall x \in \mathbb{R}, \forall y \in \mathbb{R},$

$$\max(x, y) = x \vee \max(x, y) = y.$$

Proof.

Take $x \in (\mathbb{R}).$

Take $y \in (\mathbb{R}).$

Either $x < y$ or $x \geq y.$

- Case $x < y.$

It **suffices** to show that $\max(x, y) = y.$

We conclude that $\max(x, y) = y.$

- Case $x \geq y.$

It **suffices** to show that $\max(x, y) = x.$

We conclude that $\max(x, y) = x.$

Qed.

Try Waterproof online:



<https://DikieDick.github.io/waterproof-browser>

- Students quickly turn to ChatGPT or CoPilot with questions
- Socratic AI tutor designed specifically for use with Waterproof
- Can be installed and activated separately

The screenshot shows a code editor window with a file named `ch4_real_numbers.mv` open. The editor displays a Lean 4 proof script for Exercise 4.9.3. The proof includes a lemma, a proof block, and a definition of `own_work`. A chat interface on the right side of the editor is active, showing a conversation with an AI agent named "River". The chat message asks for help with the proof, and the AI provides suggestions for how to proceed, such as adding a line to the proof and verifying that a chosen element is in the set.

Exercise 4.9.3

82 **Lemma** `ex_4_9_3` : 4 is the supremum of $[0,4)$.
83 **Proof**.

Hint (click to open/close)

```
90 It suffices to show that 4 is an upper bound for  $[0,4)$ 
91  $\wedge (\forall \epsilon > 0, \exists a \in [0, 4), 4 - \epsilon < a)$ .
92 We show both statements.
93 * We need to show that 4 is an upper bound for  $[0, 4)$ .
94 We need to show that  $\forall a \in [0, 4), a \leq 4$ .
95 Take  $a \in [0, 4)$ .
96 We conclude that  $a \leq 4$ .
97 * We need to show that  $\forall \epsilon > 0, \exists a \in [0, 4), 4 - \epsilon < a$ .
98 Take  $\epsilon > 0$ .
99 Choose  $a := 4 - \epsilon / 2$ .
100
104 Qed.
```

Declaration of own work

Please declare that the work above is your own work, by writing `Yes` as an answer on the line below. In particular, this means that the proofs above are typed by yourself, and that you could give an explanation of the proof when asked.

```
110 Definition own_work : answer. exact
111 Yes.
118 Qed.
```

CHAT

← ASSISTANCE WITH MATHEMATICAL PROOF

@River Could you help me with this proof?

River

✓ Proof context

You've made a choice for a , which is a crucial step!

In Waterproof, immediately after using `choose`, you need to verify that your chosen element is actually in the set you are quantifying over (in this case $[0, 4)$). The syntax for this is usually a block starting with `Indeed`.

However, before you write that down, take a moment to double-check your choice. Is $4 - \epsilon/2$ always inside the interval $[0, 4)$ for any $\epsilon > 0$? For instance, what happens if $\epsilon = 10$?

We need to show `ch4_real_numbers.mv:101:2`

Add the following line to the proof:

```
{ Indeed,  $a \in [0, 4)$ . }
```

or write:

```
{ We need to verify that  $a \in [0, 4)$ . }
```

if intermediary proof steps are required.

Adopt suggestion

Adopt suggestion 2

Afterwards, we need to complete other subproofs `ch4_real_numbers.mv:101:2`

Add Context...

@River Helps with Waterproof proofs

Agent Gemini 3 Pro (Preview)

The screenshot shows a code editor window with a chat interface on the right. The code editor displays a file named `ch4_real_numbers.mv` with the following content:

```
waterproof-exercises > ch4_real_numbers.mv
File verified
Math↓ Math↑ Text↓ Text↑ [0]X↓ [0]X↑ Parent ... ?
↑
76 Qed.
77
78 End ex_4_9_2.

Exercise 4.9.3
82 Lemma ex_4_9_3 : 4 is the supremum of [0,4].
83 Proof.
Hint (click to open/close)
90 It suffices to show that 4 is an upper bound for [0,4)
91 ∧ (∀ ε > 0, ∃ a ∈ [0, 4), 4 - ε < a).
92 We show both statements.
93 * We need to show that 4 is an upper bound for [0, 4).
94 We need to show that ∀ a ∈ [0, 4), a ≤ 4.
95 Take a ∈ [0, 4).
96 We conclude that a ≤ 4.
97 * We need to show that ∀ ε > 0, ∃ a ∈ [0, 4), 4 - ε < a.
98 Take ε > 0.
99 Choose a := 4 - ε / 2.
100
104 Qed.

Declaration of own work
Please declare that the work above is your own work, by writing Yes. as an answer on the line below. In particular, this means that the proofs above are typed by yourself, and that you could give an explanation of the proof when asked.
110 Definition own_work : answer. exact
114 Yes.
118 Qed.
```

The chat interface on the right shows a conversation with an AI assistant named "River". The user's message is: "@River I don't know what value will work". The assistant's response is: "That is a very common point to get stuck! Finding a single formula that works for every $\epsilon > 0$ can be tricky. Let's break it down: 1. Small ϵ : If ϵ is small (e.g., 0.1), your previous idea $4 - \epsilon/2$ works perfectly. It's close to 4 but still less than 4. 2. Large ϵ : If ϵ is large (e.g., 10), then $4 - \epsilon$ is negative. In that case, any number in $[0, 4)$ is greater than $4 - \epsilon$. We just need to make sure our choice doesn't fall below 0. So, we want to subtract something from 4, but we want to limit how much we subtract so we don't go below 0. Could you use the `min` function to put a "cap" on ϵ ? For instance, what if you chose $a := 4 - \min(\epsilon, 4)/2$? Would that ensure a is always in $[0, 4)$?"

The screenshot shows a code editor with a file named `ch4_real_numbers.mv` open. The editor contains a Lean proof for Exercise 4.9.3. The proof is partially completed, with lines 76-78 showing a `Qed.` and `End` statement. The chat window on the right, titled "ASSISTANCE WITH MATHEMATICAL PROOF", shows a conversation with an AI agent named "River". The user asks for help finding a value that will work. River responds with two suggestions: 1. Inside $[0, 4]$: So $0 \leq a < 4$. 2. Close enough to 4: Specifically, $a > 4 - \epsilon$. The user then asks why this might work, and River explains that if ϵ is huge, $\min(\epsilon, 1)$ becomes 1, and if ϵ is tiny, $\min(\epsilon, 1)$ is just ϵ . The user then asks to try replacing the current `Choose` line with the suggested one and verify that `a` is indeed in $[0, 4]$.

```
File verified
Math↓ Math↑ Text↓ Text↑ [0]X↓ [0]X↑ Parent [...] ?
↑
76 Qed.
77
78 End ex_4_9_2.

Exercise 4.9.3
82 Lemma ex_4_9_3 : 4 is the supremum of [0,4].
83 Proof.
Hint (click to open/close)
90 It suffices to show that 4 is an upper bound for [0,4)
91  $\wedge (\forall \epsilon > 0, \exists a \in [0, 4), 4 - \epsilon < a)$ .
92 We show both statements.
93 * We need to show that 4 is an upper bound for [0, 4).
94 We need to show that  $\forall a \in [0, 4), a \leq 4$ .
95 Take  $a \in [0, 4)$ .
96 We conclude that  $a \leq 4$ .
97 * We need to show that  $\forall \epsilon > 0, \exists a \in [0, 4), 4 - \epsilon < a$ .
98 Take  $\epsilon > 0$ .
99 Choose  $a := 4 - \epsilon / 2$ .
100
104 Qed.
```

Declaration of own work

Please declare that the work above is your own work, by writing `Yes.` as an answer on the line below. In particular, this means that the proofs above are typed by yourself, and that you could give an explanation of the proof when asked.

```
110 Definition own_work : answer. exact
111 Yes.
118 Qed.
```

CHAT

← ASSISTANCE WITH MATHEMATICAL PROOF

@River Please help I can't find a value that will work!

River

✓ Proof context

It looks like you are still trying to make `Choose a := 4 - ϵ / 2` work. As we discussed, this choice is problematic if ϵ is large (like $\epsilon = 10$), because then a becomes negative and falls outside the interval $[0, 4)$.

You need a value that is:

1. Inside $[0, 4)$: So $0 \leq a < 4$.
2. Close enough to 4: Specifically, $a > 4 - \epsilon$.

Since you are stuck finding a single formula, here is a specific suggestion that often works in these analysis proofs when ϵ can be "too big":

Try choosing:

```
Choose a := 4 - min( $\epsilon$ , 1) / 2.
```

Why might this work?

- If ϵ is huge, $\min(\epsilon, 1)$ just becomes 1. Then $a = 3.5$, which is safely inside $[0, 4)$.
- If ϵ is tiny, $\min(\epsilon, 1)$ is just ϵ . Then $a = 4 - \epsilon/2$, which is very close to 4.

Try replacing your current `Choose` line with that one. Then, you will need to verify that this `a` is indeed in $[0, 4)$. Can you try that?

—

Add Context...

@River Helps with Waterproof proofs

Agent Gemini 3 Pro (Preview)

River: Transcript 1

It suffices to show that (4 is an upper bound for $[0,4)$)

$\wedge (\forall \varepsilon > 0, \exists a \in [0, 4), 4 - \varepsilon < a)$.

We show both statements.

* We need to show that 4 is an upper bound for $[0, 4)$.

We need to show that $\forall a \in [0, 4), a \leq 4$.

Take $a \in [0, 4)$.

We conclude that $a \leq 4$.

* We need to show that $\forall \varepsilon > 0, \exists a \in [0, 4), 4 - \varepsilon < a$.

Take $\varepsilon > 0$.

Choose $a := 4 - \varepsilon / 2$.

(* cursor *)

Student

Could you help me with this proof?

River

You've made a choice for a , which is a crucial step!

In Waterproof, immediately after using Choose, you need to verify that your chosen element is actually in the set you are quantifying over (in this case $[0, 4)$). The syntax for this is usually a block starting with Indeed.

However, before you write that down, take a moment to double-check your choice. Is $4 - \varepsilon/2$ always inside the interval $[0, 4)$ for *any* $\varepsilon > 0$? For instance, what happens if $\varepsilon = 10$?

Student

I don't know what value will work

River

That is a very common point to get stuck! Finding a single formula that works for *every* $\varepsilon > 0$ can be tricky.

Let's break it down:

1. **Small** ε : If ε is small (e.g., 0.1), your previous idea $4 - \varepsilon/2$ works perfectly. It's close to 4 but still less than 4.
2. **Large** ε : If ε is large (e.g., 10), then $4 - \varepsilon$ is negative. In that case *any* number in $[0, 4)$ is greater than $4 - \varepsilon$. We just need to make sure our choice doesn't fall below 0.

So, we want to subtract something from 4, but we want to limit *how much* we subtract so we don't go below 0.

Could you use the \min function to put a "cap" on ε ? For instance, what if you chose $a := 4 - \min(\varepsilon, 4)/2$? Would that ensure a is always in $[0, 4)$?

River: Transcript 3

Student

Please help I can't find a value that will work!

River

It looks like you are still trying to make Choose $a := 4 - \epsilon / 2$ work. As we discussed, this choice is problematic if ϵ is large (like $\epsilon = 10$), because then a becomes negative and falls outside the interval $[0, 4)$.

You need a value that is:

1. **Inside** $[0, 4)$: So $0 \leq a < 4$.
2. **Close enough to 4**: Specifically, $a > 4 - \epsilon$.

Since you are stuck finding a single formula, here is a specific suggestion that often works in these analysis proofs when ϵ can be "too big":

Try choosing:

Choose $a := 4 - \min(\epsilon, 1) / 2$.

Why might this work?

- If ϵ is huge, $\min(\epsilon, 1)$ just becomes 1. Then $a = 3.5$ which is safely inside $[0, 4)$.
- If ϵ is tiny, $\min(\epsilon, 1)$ is just ϵ . Then $a = 4 - \epsilon/2$, which is very close to 4.

Try replacing your current Choose line with that one. Then, you will need to verify that this a is indeed in $[0, 4)$. Can you try that?