Formalisation of constructible numbers

Ludwig Monnerjahn

July 15, 2024

This talk is about formalising the proof that the set of all constructible points $\mathcal{M}_\infty$ forms a field. This is the first step needed to solve ancient construction problems, such as doubling the cube and trisecting an angle. $\mathcal{M}_\infty$ is a subset of the complex numbers $\mathbb{C}$. Therefore, it is sufficient to demonstrate that $\mathcal{M}_\infty$is a subfield of $\mathbb{C}$. In lean we do this by defining a structure on $\mathcal{M}_\infty$.

```
noncomputable def MFinf : Subfield ℂ where
    carrier := _
    zero_mem' := _
    one_mem' := _
    add_mem' := _
    neg_mem' := _
    mul_mem' := _
    inv_mem' := _
```

The next step is to complete the blanks. This will entail first filling in the carrier set of $\mathcal{M}_\infty$. To do this, it is first necessary to recall the definitions of $\mathcal{M}_\infty$ and state them in lean.

1 Definition of $\mathcal{M}_\infty$

We start with a basic set of points $\mathcal{M} \subseteq \mathbb{C}$ in the complex plane.

Definition 1.1 (Line)**.** A line l through two points $x, y \in \mathbb{C}$ with $x \neq y$ is defined by the set:

$$l := \{\lambda x + (1 - \lambda)y \mid \lambda \in \mathbb{R}\}.$$

```
structure line where
    (z₁ z₂ : ℂ)

def line.points (l: line) : Set ℂ:=
    {(t : ℂ) * l.z₁ + (1-t) * l.z₂ | (t : ℝ)}
```

Definition 1.2 (Circle)**.** A circle c with center $z \in \mathbb{C}$ and radius $r \in \mathbb{R}_{\geq 0}$ is defined by the set:

$$c := \{z \in \mathbb{C} \mid \|z - c\| = r\}.$$

```
structure circle where
    (c : ℂ)
    (r : ℝ)

def circle.points (c: circle) := Metric.sphere c.c c.r
noncomputable def circle.points' (c: circle) :=
    (⟨c.c, c.r⟩ : EuclideanGeometry.Sphere ℂ)
```

Definition 1.3 (Set of lines and circles)**.** $\mathcal{L}(\mathcal{M})$ is the set of all real straight lines defined by two points in $\mathcal{M}$.
And $\mathcal{C}(\mathcal{M})$ is the set of all circles defined by a center in $\mathcal{M}$ and a radius equal to the distance between two points in $\mathcal{M}$.

```
def L (M:Set ℂ): Set line := {l |∃ z₁ z₂, l = {z₁ := z₁, z₂ := z₂} ∧
    z₁ ∈  M ∧ z₂ ∈ M ∧ z₁ ≠ z₂}
def C (M:Set ℂ): Set circle := {c |∃ z r₁ r₂, c = {c:=z, r:=(dist
    r₁ r₂)} ∧ z ∈ M ∧ r₁ ∈ M ∧ r₂ ∈ M}
```

Definition 1.4 (Rules to construct a point)**.** We define operations that can be used to construct new points.

1. (ILL) is the intersection of two different lines in $\mathcal{L}(\mathcal{M})$.
2. (ILC) is the intersection of a line in $\mathcal{L}(\mathcal{M})$ and a circle in $\mathcal{C}(\mathcal{M})$.
3. (ICC) is the intersection of two different circles in $\mathcal{C}(\mathcal{M})$.

$ICL(\mathcal{M})$ is the set $\mathcal{M}$ combined with all points that can be constructed using the operations (ILL), (ILC) and (ICC).

```
def ill (M:Set ℂ): Set ℂ := { z  |∃l₁ ∈ L M, ∃ l₂ ∈ L M,  z ∈
    l₁.points ∩ l₂.points ∧ l₁.points ≠ l₂.points}
def ilc (M:Set ℂ): Set ℂ := { z  |∃c ∈ C M, ∃ l ∈ L M,  z ∈
    c.points ∩ l.points}
def icc (M:Set ℂ): Set ℂ := { z  |∃c₁ ∈ C M, ∃ c₂ ∈ C M,  z ∈
    c₁.points ∩ c₂.points ∧ c₁.points' ≠ c₂.points'}

def ICL_M (M : Set ℂ) : Set ℂ := M ∪ ill M ∪ ilc M ∪ icc M
```

Definition 1.5 (Set of constructible points)**.** We define inductively the chain

$$\mathcal{M}_0 \subseteq \mathcal{M}_1 \subseteq \mathcal{M}_2 \subseteq \dots$$

with $\mathcal{M}_0 = \mathcal{M}$ and $\mathcal{M}_{n+1} = ICL(\mathcal{M}_n)$.
And call $\mathcal{M}_\infty = \bigcup_{n\in\mathbb{N}} \mathcal{M}_n$ the set of all constructable points.

```
def M_I (M : Set ℂ) : ℕ → Set ℂ
    | 0 => M
    | (Nat.succ n) => ICL_M (M_I M n)

def M_inf (M : Set ℂ) : Set ℂ :=  ⋃ (n : ℕ), M_I M n
```

We can now fill in the first blank:

```
noncomputable def MFinf (M: Set ℂ) : Subfield ℂ where
    carrier := M_inf M
    ...
```

2 Zero and one in $\mathcal{M}_\infty$

Without loss of generality we can assume that $\mathcal{M}$ contains the points 0 and 1. Because constructing with less than two points is trivial ($\mathcal{M} = ILC(\mathcal{M}$ and therefore $\mathcal{M} = \mathcal{M}_\infty$)) and we can always scale and translate the plane to get 0 and 1 in $\mathcal{M}$. And since we assume that $\mathcal{M}$ contains the points 0 and 1 we can fill in the next two blanks, after proving that $\mathcal{M} \subseteq \mathcal{M}_\infty$.

Lemma 2.1 ($\mathcal{M} \subseteq \mathcal{M}_i$)**.** The set $\mathcal{M}$ is contained in $\mathcal{M}_i$, i.e. $\mathcal{M} \subseteq \mathcal{M}_i$.

Proof. Combining the fact that $\mathcal{M}_0 = \mathcal{M}$ 1.5 and the monotony of $\mathcal{M}_i$ which follows from $\mathcal{M} \subset \mathcal{ICL}(\mathcal{M})$. □

```
lemma M_in_ICL_M (M : Set ℂ) : M ⊆ ICL_M M := by
    unfold ICL_M
    intro x hx
    left; left; left
    exact hx

lemma M_I_Monotone (M : Set ℂ) : ∀n, M_I M n ⊆ M_I M (n+1) := by
    intro n
    apply M_in_ICL_M

lemma M_in_M_I (M : Set ℂ) : ∀n, M ⊆ M_I M n := by
    intro n
    induction n
    simp only [M_I]
    exact fun ⦃a⦄ a => a
    case succ n hn =>
        apply le_trans hn
        apply M_I_Monotone
```

Lemma 2.2 ($\mathcal{M}_i \subseteq \mathcal{M}_\infty$)**. ??** The set $\mathcal{M}_i$ is contained in $\mathcal{M}_\infty$, i.e. $\mathcal{M}_i \subseteq \mathcal{M}_\infty$.

Proof. Follows from the definition of $\mathcal{M}_\infty$. □

```
lemma M_I_in_M_inf (M : Set ℂ)(m: ℕ): M_I M m ⊆ M_inf M := by
    refine Set.subset_iUnion_of_subset m fun ⦃a⦄ a => a
```

Lemma 2.3 ($\mathcal{M} \subseteq \mathcal{M}_\infty$)**.** The set $\mathcal{M}$ is contained in $\mathcal{M}_\infty$.

Proof. Combining $\mathcal{M} \subseteq \mathcal{M}_i$ 2.1 and $\mathcal{M}_i \subseteq \mathcal{M}_\infty$ **??** we get the result. □

```
lemma M_M_inf (M : Set ℂ) : M ⊆ M_inf M := by
    apply le_trans (M_in_M_I M 0) (M_I_in_M_inf M 0)6
```

By applying this, the following can be obtained:

```
noncomputable def MField (M: Set ℂ)(h₀: 0 ∈ M)(h₁: 1∈ M):
        Subfield ℂ where
    carrier := M_inf M
    zero_mem' := by exact M_M_inf M h₀
    one_mem' := by exact M_M_inf M h₁
    ...
```

3 Construction

In order to complete the construction, it is necessary to define the addition, multiplication, negation and inversion of constructable numbers. The following chapter presents a proof schema, whereby lines and circles are used to demonstrate that the desired point is contained within their intersection. As the proofs are lengthy and repetitive, they have been omitted from the handout. Instead, the construction and underlying concepts are presented. The complete proofs can be found in the blueprint.

Lemma 3.1 (Addition of complex numbers)**.** For $z_1, z_2 \in M_\infty$ is $z_1 + z_2 \in M_\infty$.

This construction is taken from [2].
One can construct the point $z_1 + z_2$ by drawing a circle with center z_1 and radius $\|z_2\|$ and a circle with center z_2 and radius $\|z_1\|$ and taking the intersection of the two circles Fig.1.

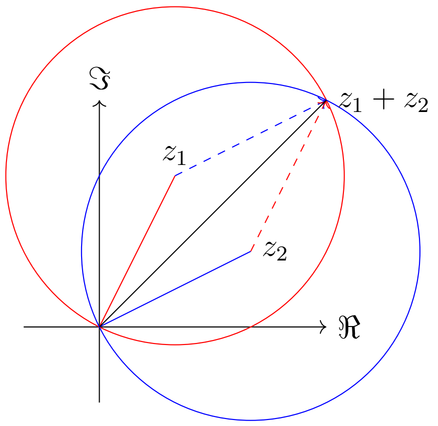

Figure 1: Construction of $z_1 + z_2$

```
lemma add_M_Inf (M: Set ℂ) (h₀: (0:ℂ)∈ M) (z₁ z₂ : ℂ) (hz₁ : z₁ ∈
    (M_inf M)) (hz₂ : z₂ ∈ (M_inf M)):
        z₁ + z₂ ∈ (M_inf M) := by
    let c₁ : Construction.circle := {c := z₁, r := (dist 0 z₂)}
    let c₂ : Construction.circle := {c := z₂, r := (dist 0 z₁)}
    let l: line := {z₁ := z₁, z₂ := 0}
    have hc₁ : c₁ ∈ C (M_inf M) := by
    refine ⟨z₁, 0, z₂, ?_, hz₁,M_M_inf M h₀, hz₂⟩
    simp [c₁]
    have hc₂ : c₂ ∈ C (M_inf M) := by
    refine ⟨z₂, 0, z₁, ?_,hz₂,M_M_inf M h₀,hz₁⟩
    simp [c₂]
    by_cases h: (z₁ = z₂)
    . by_cases hz₁0: (z₁ = 0)
    . simp only [hz₁0, zero_add, hz₂]
    . have hl : l ∈ L (M_inf M) := by
        refine ⟨z₁, 0, ?_, hz₁, M_M_inf M h₀, hz₁0⟩
        simp [l, hz₁0]
        apply ilc_M_inf M
        refine ⟨c₁, hc₁, l, hl, ⟨?_, ⟨2,?_⟩⟩⟩
        . simp [circle.points, h]
        . simp [h, two_mul]
    refine icc_M_inf M ⟨c₁,hc₁, c₂,hc₂, ?_⟩
    simp [circle.points, Set.mem_inter_iff]
    exact circle_not_eq_iff  (by exact h)
```

Lemma 3.2 (Negative complex numbers)**.** For $z \in M_\infty$, $-z$ is in M_∞.

This construction is taken from [2].
To get the point $-z$ we can use the second intersection of the line through 0 and z with circle with center 0 and radius $\|z\|$ Fig.2.

```
lemma z_neg_M_inf (M: Set ℂ) (h₀: (0:ℂ)∈ M) (z : ℂ)
        (hz : z ∈ (M_inf M)) : -z ∈ (M_inf M) := by
    by_cases z0:(z=0)
    . simp [z0, M_M_inf M h₀]
    let l : line := {z₁ := 0, z₂ := z}
    let c : Construction.circle := {c := 0, r := (dist 0 z)}
    have hl : l ∈ L (M_inf M) := by
      refine ⟨0, z, ?_, M_M_inf M h₀, hz, ?_⟩
      simp only [l]
      simp  [eq_comm, z0]
    have hc : c ∈ C (M_inf M) := by
      refine ⟨0, 0, z, ?_, M_M_inf M h₀, M_M_inf M h₀, hz⟩
      simp [l, c]
    apply ilc_M_inf M
    refine ⟨c , hc, l, hl, ?_⟩
    simp [circle.points, line.points]
```

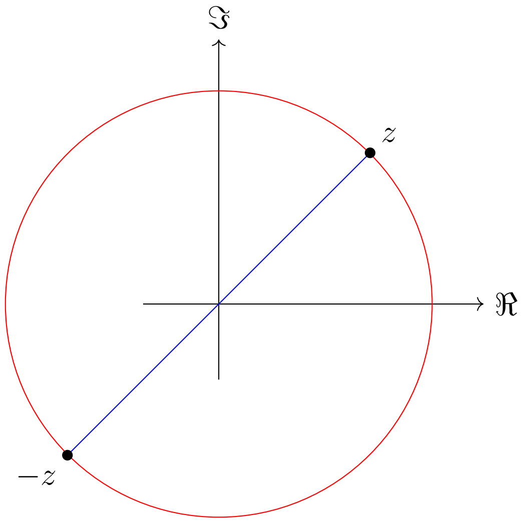

Figure 2: Construction of $-z$

```
    refine ⟨2, (by push_cast; ring_nf)⟩
```

Lemma 3.3 (Multiplication of positive real numbers)**.** For $a, b \in M_\infty \cap \mathbb{R}$, $a \cdot b \in M_\infty$.

This construction is taken from [1].
To get the point $a \cdot b$ we draw a line through a and $\imath$ and a parallel line through $\imath b$. The intersection of the second line with the real axis is $a \cdot b$ Fig.3.

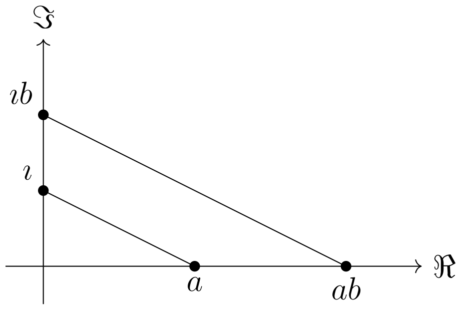

Figure 3: Construction of $z_1 \cdot z_2$

Remark. If you look at how you chose the representatives of the parallel line $x = a + In - b$ and $y = Ib$, you can prove that they are in M_∞ without the first line, so you can prove this with only two lines.

```
lemma ab_in_M_inf (M: Set ℂ) (h₀: 0 ∈ M) (h₁: 1 ∈ M) (a b :ℝ)
    (ha: ↑a ∈ M_inf M) (hb: ↑b ∈ M_inf M): ↑(a * b) ∈ M_inf M := by
  by_cases h: a*b = 0
  . rw[h]
    exact M_M_inf M h₀
  let l : line := {z₁ := a+I*b-I, z₂ := I*b}
  let lr : line := {z₁ := 1, z₂ := 0}
  have hl : l ∈ L (M_inf M) := by
    refine ⟨(a+I*b-I), I*b, (by simp), ?_, ir_M_inf _ h₀ h₁ _ hb, ?_⟩
    . simp only [sub_M_Inf, add_M_Inf, ir_M_inf M h₀ h₁ b hb,
   imath_M_inf, h₀, h₁, ha]
    simp [ext_iff]
  have hlr : lr ∈ L (M_inf M) := by
    refine ⟨1, 0, (by simp only), M_M_inf M h₁,  M_M_inf M h₀, ?_⟩
    simp only [ne_eq, one_ne_zero, not_false_eq_true]
  refine ill_M_inf M ⟨l,hl, lr, hlr, ⟨⟨b, ?_⟩, ⟨a*b, ?_⟩⟩, ?_ ⟩
  push_cast; ring_nf
  push_cast; ring_nf
  refine line_not_eq_if'  l lr ⟨0, ⟨?_, ?_⟩⟩
  . simp[line.points]
  . simp[line.points, ext_iff, sub_mul, mul_sub, sub_eq_zero]
    rw[mul_eq_zero, or_comm, Mathlib.Tactic.PushNeg.not_or_eq] at h
    exact h
```

Corollary 3.4 (Multiplication of complex numbers)**.** For $z_1, z_2 \in M_\infty$ is $z_1 \cdot z_2$ in M_∞.

Proof. Let $z_1 = a + \imath b$ and $z_2 = c + \imath d$. Then

$$z_1 \cdot z_2 = (a + \imath b) \cdot (c + \imath d) = (a \cdot c - b \cdot d) + \imath(a \cdot d + b \cdot c).$$

By combining the Lemmas 3.1, 3.3 with subtraction, real and imaginary part we get that $z_1 \cdot z_2 \in M_\infty$. □

```
lemma z_iff_re_im_M_inf (M: Set ℂ) (h₀: 0 ∈ M) (h₁: 1 ∈ M) (z: ℂ):
    z ∈ M_inf M ↔ ↑z.re ∈ M_inf M ∧ ↑z.im ∈ M_inf M := sorry
```

```
lemma mul_M_inf (M: Set ℂ) (h₀: 0 ∈ M) (h₁: 1 ∈ M) (a b :ℂ )
    (ha: a ∈ M_inf M) (hb: b ∈ M_inf M): a * b ∈ M_inf M:= by
  refine (z_iff_re_im_M_inf M h₀ h₁ (a * b)).mpr ⟨?_, ?_⟩ <;>
  simp only [mul_re, mul_im, ofReal_sub, ofReal_add]
  . apply sub_M_Inf M h₀
    exact ab_in_M_inf M h₀ h₁ _ _ (real_in_M_inf M h₀ h₁ a ha)
        (real_in_M_inf M h₀ h₁ b hb)
    exact ab_in_M_inf M h₀ h₁ _ _ (im_in_M_inf M h₀ h₁ a ha)
        (im_in_M_inf M h₀ h₁ b hb)
  . apply add_M_Inf M h₀
    exact ab_in_M_inf M h₀ h₁ _ _ (real_in_M_inf M h₀ h₁ a ha)
        (im_in_M_inf M h₀ h₁ b hb)
    exact ab_in_M_inf M h₀ h₁ _ _ (im_in_M_inf M h₀ h₁ a ha)
        (real_in_M_inf M h₀ h₁ b hb)
```

Lemma 3.5 (Inverse of a pos real number)**.** If $a \in M_\infty \cap \mathbb{R}$, then a^{-1} is in M_∞.

This can be constructed analog to the multiplication of positive real numbers. Using the fact that $a \cdot a^{-1} = 1$. Draw a line through 1 and $\imath a$ and a parallel line through $\imath$. The intersection of the second line with the real axis is a^{-1} Fig.4.

Proof. Without loss of generality we can assume that $a \neq 0$.
Then the proof is analog to the proof of Lemma 3.3, we just need two lines $l = \{1 - \imath a + \imath, \imath\}$ and $l_\Re = \{1, 0\}$.
That there are in $\mathcal{L}(\mathcal{M}_\infty)$ follows analog to the proof of Lemma 3.3.
So we have just to show that $a^{-1} \in l$, i.e. $\exists t : t(1 - \imath a + \imath) + (1 - t)I = a^{-1}$

$$t(1 - \imath a + \imath) + (1 - t)\imath \overset{t:=a^{-1}}{=} a^{-1} - a^{-1}\imath a + a^{-1}\imath + \imath - a^{-1}\imath = a^{-1}.$$

The rest follws analog. □

```
lemma ainv_in_M_inf (M: Set ℂ) (h₀: 0 ∈ M) (h₁: 1 ∈ M) (a :ℝ)
        (ha: ↑a ∈ M_inf M): ↑(a⁻¹) ∈ M_inf M := by
    by_cases h: a = 0
    . simp [h]
      exact M_M_inf _ h₀
    let l: line := {z₁ := 1-I*a+I, z₂ := I}
```

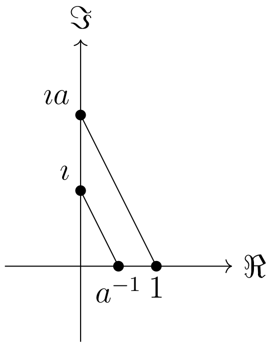

Figure 4: Construction of z^{-1}

```
let lr : line := {z₁ := 1, z₂ := 0}
have hl : l ∈ L (M_inf M) := by
  refine ⟨(1-I*a+I), I, (by simp), ?_, imath_M_inf M h₀ h₁, ?_⟩
  . apply add_M_Inf M h₀ (1-I*a) I ?_ (imath_M_inf M h₀ h₁)
    exact sub_M_Inf M h₀ 1 (I*a) (M_M_inf M h₁)
        (mul_M_inf M h₀ h₁ _ _ (imath_M_inf M h₀ h₁) ha)
  simp [ext_iff]
have hlr : lr ∈ L (M_inf M) := by
  refine ⟨1, 0, (by simp), M_M_inf M h₁, M_M_inf M h₀, ?_⟩
  simp
refine ill_M_inf M ⟨l, hl, lr, hlr, ⟨⟨a⁻¹, ?_⟩ , ⟨a⁻¹, ?_⟩⟩, ?_⟩
. ring_nf
simp [h, mul_rotate]
. simp only [ofReal_inv, mul_one, mul_zero, add_zero]
refine line_not_eq_if' l lr ⟨0, ⟨?_, ?_⟩⟩
. simp [line.points]
. simp [line.points, ext_iff]
```

Remark. The non-terminal `simp` and the rest without `only` are only used for better readability.

Corollary 3.6 (Inverse of a complex number)**.** If $z \in M_\infty$, then z^{-1} is in M_∞.

Proof. For $z \in M_\infty$ we can write $z = a + \imath b$ with $a, b \in \mathbb{R}$. Then

$$z^{-1} = \frac{1}{z} = \frac{\overline{z}}{z\overline{z}} = \frac{a - \imath b}{a^2 + b^2} = (a - \imath b) \cdot (aa + bb)^{-1}.$$

Now we can again combine the lemmas for addition 3.1, subtraction [blue print], multiplication 3.4 and the corollary for the inverse of a positive real number 3.5 with the exists of real an imaginary [blue print] part to conclude that $z^{-1} \in M_\infty$. □

```
lemma z_inv_eq (z:ℂ) (hz: z ≠ 0): z⁻¹ = z.re /
    (z.re^2+z.im^2)-(z.im/ (z.re^2+z.im^2) )*I := sorry

lemma inv_M_inf (M: Set ℂ) (h₀: 0 ∈ M) (h₁: 1 ∈ M) (a :ℂ )
        (ha: a ∈ M_inf M): a⁻¹ ∈ M_inf M:= by
    by_cases h: a = 0
    . simp only [h, inv_zero]
    exact M_M_inf _ h₀
    simp_rw [z_inv_eq _ h, Field.div_eq_mul_inv, pow_two]
    apply sub_M_Inf M h₀
    . apply mul_M_inf M h₀ h₁ _ _ (real_in_M_inf M h₀ h₁ a ha)
    norm_cast
    apply ainv_in_M_inf M h₀ h₁
    push_cast
    apply add_M_Inf M h₀
    exact mul_M_inf M h₀ h₁ _ _ (real_in_M_inf M h₀ h₁ _ ha)
        (real_in_M_inf M h₀ h₁ _ ha)
    exact mul_M_inf M h₀ h₁ _ _ (im_in_M_inf M h₀ h₁ _ ha)
        (im_in_M_inf M h₀ h₁ _ ha)
    . apply mul_M_inf M h₀ h₁ _ _ ?_ (imath_M_inf M h₀ h₁)
    apply mul_M_inf M h₀ h₁ _ _ (im_in_M_inf M h₀ h₁ _ ha)
    norm_cast
    apply ainv_in_M_inf M h₀ h₁
    push_cast
    apply add_M_Inf M h₀
    exact mul_M_inf M h₀ h₁ _ _ (real_in_M_inf M h₀ h₁ _ ha)
        (real_in_M_inf M h₀ h₁ _ ha)
    exact mul_M_inf M h₀ h₁ _ _ (im_in_M_inf M h₀ h₁ _ ha)
        (im_in_M_inf M h₀ h₁ _ ha)
```

4 Conclusion

At last, we have assembled the requisite elements for the construction of the field of constructible numbers $\mathcal{M}_\infty$.

```
noncomputable def MField (M: Set ℂ)(h_0: 0 ∈ M)(h_1: 1∈ M):
        Subfield ℂ where
    carrier := M_inf M
    zero_mem' := by exact M_M_inf M h_0
    one_mem' := by exact M_M_inf M h_1
    add_mem' := by apply add_M_Inf M h_0
    neg_mem' := by apply z_neg_M_inf M h_0
    mul_mem' := by apply mul_M_inf M h_0 h_1
    inv_mem' := by apply inv_M_inf M h_0 h_1
```

Now it is just an `instance`, proven by `exact?`. To get the structure of the field. Normally this would be done by `infer_instance`, but I want to show the proof in this talk.

```
noncomputable instance MField_field (M: Set ℂ)(h_0: 0 ∈ M)
        (h_1: 1∈ M): Field (MField M h_0 h_1) := by
    exact SubfieldClass.toField (Subfield ℂ) (MField M h_0 h_1)
```

This can be used to proof that $x \in \mathbb{C}$ is in $\mathcal{M}_\infty$ if and only if the degree of x over $\mathbb{Q}(M)$ is of the form 2^n for some $n \in \mathbb{N}$.

References

[1] D.A. Cox. *Galois Theory*. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley, 2012.

[2] JAN SCHRÖER. Einführung in die algebra. SKRIPT, WS 22/23, BONN, 2023.