

虚数って存在するの？

西 航

2021/10/19

はじめに

質問する奴は偉い

- 初めに質問する奴は偉い
 - 次の人が続きやすくなる
 - 馬鹿な質問をする奴は偉い
 - 質問の内容のハードルを下げる
 - 関係ない質問をする奴は偉い
 - 話が広がる
-
- <https://togetter.com/li/1116798> からパクりました。
 - 完全にその通りだと思います。
 - 今回、ところどころで質問タイムも設けるつもりですが、それに関係なく、いつでも何でも聞いてください。
 - 時間の都合で拾いきれない場合などは、回答は後日になるかもしれませんが、そういう場合であっても質問は大歓迎です。

目次

- ① $\sqrt{-1}$ について雑多な話
 - みんなぶっちゃけどう思ってるの？
 - あったりなかったりする $\sqrt{-1}$
- ② 複素数を使うと嬉しいという話
 - 3 次方程式の解の公式
 - ガウス整数
- ③ 数は作れるという話
 - interface と数
 - 自然数から整数を作る
 - 整数から複素数を作る
- ④ まとめ

いくつかの質問

質問 1

$\sqrt{-1}$ は存在するか？

いくつかの質問

質問 1

$\sqrt{-1}$ は存在するか？

質問 2

-1 は存在するか？

いくつかの質問

質問 1

$\sqrt{-1}$ は存在するか？

質問 2

-1 は存在するか？

質問 3

1 は存在するか？

いくつかの質問

質問 1

$\sqrt{-1}$ は存在するか？

質問 2

-1 は存在するか？

質問 3

1 は存在するか？

（筆者なりの）回答

現代の数学は、これらの質問に答える必要がないように構成されている。ただし、どれか 1 つの存在を認めるのであれば、ほかの 2 つについても認めるべき。

もう1つ質問

質問

数とは何か？

もう1つ質問

質問

数とは何か？

（筆者なりの）回答

現代の数学は、この質問にも答える必要がないように構成されている。強いて言えば、そのとき数だと思いたいものを数と呼べばよい。ただし、

- 自然数とは何か？
- 実数とは何か？

という質問であれば、ある程度決まった答えがある。

$\sqrt{-1}$ はあったりなかったりする？

- $\sqrt{-1}$ は、ある意味では存在します。
- あなたは、マジで 5 で割った余りにしか興味がないとしましょう。
 - そんなわけねえだろ、と思うかもしれませんが、そういう設定でお願いします。
- あなたにとって整数とは、0, 1, 2, 3, 4 だけのことです。
 - 6 や 11 や 21 といった整数は、あなたにとってはすべて 1 です。マジで 5 で割った余りにしか興味がないので。
- あなたは、整数の足し算や掛け算ができます。たとえば、 $3 \times 4 = 2$ です。
 - 普通の人には 12 というのですが、あなたにとってそれは 2 です。

マジで5で割った余りにしか興味がない人の話

- -1 というのは、1を足して0になる数のこと。つまり4です。
- $\sqrt{-1}$ というのは、2乗して4になる数のこと。つまり2です。
- あなたにとって、 $\sqrt{-1}$ は存在します。2のことですよね。
 - 実は、3のことでもあります。
 - 1つに決まらないのはどうなんだという気もしますが、あるのかないのかでいうと、あるということですね。

マジで7で割った余りにしか興味がない人の話

- 続いて、マジで7で割った余りにしか興味がないとしましょう。
- あなたにとって整数とは、0, 1, 2, 3, 4, 5, 6 だけのことです。
- あなたは、整数の足し算や掛け算ができます。たとえば、 $3 \times 4 = 5$ です。

マジで7で割った余りにしか興味がない人の話

- 続いて、マジで7で割った余りにしか興味がないとしましょう。
- あなたにとって整数とは、0, 1, 2, 3, 4, 5, 6 だけのことです。
- あなたは、整数の足し算や掛け算ができます。たとえば、 $3 \times 4 = 5$ です。
- -1 というのは、1 を足して 0 になる数のこと。つまり 6 です。
- $\sqrt{-1}$ というのは、2 乗して 6 になる数のことですが...
- 0, 1, 2, 3, 4, 5, 6 のどれも、2 乗して 6 にはなりません。
- $\sqrt{-1}$ は存在しませんね。

いったんまとめ

- 現代の数学は、 $\sqrt{-1}$ は存在するか？数とは何か？といった質問に答える必要がないように構成されている。
- $\sqrt{-1}$ が存在することが証明できるようなセッティングもあるし、存在しないことが証明できるようなセッティングもある。

この章はこれで終わり

質問などあればいつでもどうぞ！

2 次方程式の解の公式

次の事実は、おそらく一度は勉強したことがあると思います。

2 次方程式の解の公式

方程式

$$x^2 + ax + b = 0$$

の解は、

$$x = \frac{-a \pm \sqrt{a^2 - 4b}}{2}$$

である。

この公式で $\sqrt{\quad}$ の中身が負になる場合、現代的には「複素数解が存在する」と思ってよいわけですが、歴史的には、なかなかそうはならなかったようです。

それは当然といえば当然で、複素数に慣れていない人にとっては「複素数解がある」と考えるほうが非常識、解なしとするのが常識的でしょう。

3 次方程式の解の公式

では、次の公式はご存知でしょうか。

3 次方程式の解の公式

方程式

$$x^3 + ax^2 + bx + c = 0$$

の解の 1 つは、

$$x = \sqrt[3]{\frac{q}{2} + \frac{\sqrt{27q^2 + 4p^3}}{6\sqrt{3}}} + \sqrt[3]{\frac{q}{2} - \frac{\sqrt{27q^2 + 4p^3}}{6\sqrt{3}}} - \frac{a}{3}$$

である。ただし、 p, q は a, b, c を使って

$$p = \frac{-a^2}{3} + b, q = \frac{2a^3}{27} - \frac{ab}{3} + c$$

で与えられる数とする。

3 次方程式の解の公式

公式は知らなくても全く問題ないですが、とにかく、この公式を使って

$$x^3 - 15x - 4 = 0$$

を解いてみましょう。途中式は省略しますが、計算すると

$$x = \sqrt[3]{2 + 11\sqrt{-1}} + \sqrt[3]{2 - 11\sqrt{-1}}$$

となります。

3 次方程式の解の公式

公式は知らなくても全く問題ないですが、とにかく、この公式を使って

$$x^3 - 15x - 4 = 0$$

を解いてみましょう。途中式は省略しますが、計算すると

$$x = \sqrt[3]{2 + 11\sqrt{-1}} + \sqrt[3]{2 - 11\sqrt{-1}}$$

となります。

$(2 \pm \sqrt{-1})^3 = 2 \pm 11\sqrt{-1}$ なので、結局

$$x = (2 + \sqrt{-1}) + (2 - \sqrt{-1}) = 4$$

です。実際、

$$4^3 - 15 \times 4 - 4 = 0$$

なので、この公式が正しいことがわかります。

3 次方程式の解の公式

- 現代の人間としては、これで 3 次方程式を解くことができた、めでたしめでたし、となるわけですが...
- この公式（を、この方程式にあてはめた結果）は、発見された 16 世紀当時には、かなり「ヤバい」方法とされたようです。
- $x = \sqrt[3]{2 + 11\sqrt{-1}} + \sqrt[3]{2 - 11\sqrt{-1}} = 4$

3 次方程式の解の公式

- 現代の人間としては、これで 3 次方程式を解くことができた、めでたしめでたし、となるわけですが...
- この公式（を、この方程式にあてはめた結果）は、発見された 16 世紀当時には、かなり「ヤバい」方法とされたようです。
- $x = \sqrt[3]{2 + 11\sqrt{-1}} + \sqrt[3]{2 - 11\sqrt{-1}} = 4$
- 解は結果的にはちゃんと求まっているけど、**負の数の平方根とかいうあり得ないもの**を経由しとるやんけ !!!

3 次方程式の解の公式

- 現代の人間としては、これで 3 次方程式を解くことができた、めでたしめでたし、となるわけですが...
- この公式（を、この方程式にあてはめた結果）は、発見された 16 世紀当時には、かなり「ヤバい」方法とされたようです。
- $x = \sqrt[3]{2 + 11\sqrt{-1}} + \sqrt[3]{2 - 11\sqrt{-1}} = 4$
- 解は結果的にはちゃんと求まっているけど、**負の数の平方根とかいうあり得ないもの**を経由しとるやんけ !!!
- 歴史には詳しくないですが、どうやら当時（というか 17 世紀くらいまでのヨーロッパで）は、負の数にすら抵抗があったようです。
- でも 3 次方程式が解けないよりは、解けるほうがどう考えてもいいですね。
- これは、「虚数を数学的な考察の対象から外してしまうことがもったいない理由」の 1 つになりそうです。

素数クイズ

突然ですが、次の問題を考えてみてください。

問題

2501 は素数か？

素数クイズ

突然ですが、次の問題を考えてみてください。

問題

2501 は素数か？

次の等式がヒントです。

ヒント

$$2501 = 50^2 + 1^2 = 49^2 + 10^2$$

このヒントでわかる人は、今日の話聞かなくてもよさそうな気がしますが...

答え

答え

2501 は素数ではない。

- $2501 = 41 \times 61$ です。
- 41 と 61 は素数なので、暗算で見つけるのはそれほど簡単ではないと思います。
- $2501 = 51^2 - 10^2$ に気づけば、数秒で答えられた人もいたかも？
 - $51^2 - 10^2 = (51 + 10)(51 - 10)$ ですね。
 - 一般に $a^2 - b^2 = (a + b)(a - b)$ です。

どのあたりがヒントになっていたか

- $2501 = 50^2 + 1^2 = 49^2 + 10^2$ ということは...
- $2501 = (50 + \sqrt{-1})(50 - \sqrt{-1}) = (49 + 10\sqrt{-1})(49 - 10\sqrt{-1})$
- ぜんぜん違う数の積に分解できる！
- 素数であれば、このような分解はありえないことが証明できます。
 - もちろんこのことを証明するために、複素数まで拡張した素因数分解を深く知る必要があります。
- 上の分解は「素因数分解」ではないのですが、ここでは
 - 何らかの数の積に分解できたこと
 - その分解した要素は、かたや $50 + \sqrt{-1}$ 、かたや $49 + 10\sqrt{-1}$ と一見して違うものであることが重要です。
- したがって、2501 は素数ではありません。

ガウス整数の素元分解

- 2501 は素数か？という、虚数なんて全く関係なさそうな整数の問題が、虚数を含めた「素因数分解」の様子を深く知ることによって解決できる
 - 正確には、このように複素数にまで拡張した整数っぽいものを「ガウス整数」といい、ガウス整数においては素因数分解ではなく「素元分解」といいます。
- このこともまた、「虚数を数学的な考察の対象から外してしまうことがもったいない理由」の1つになりそうです。

いったんまとめ

- 3 次方程式の解の公式には、実数解を表示する場合であっても、負の数の平方根が現れることがある。
- 整数の性質は、整数の概念を複素数にまで拡張したガウス整数を考えることで、より深く理解できる。

いったんまとめ

- 3 次方程式の解の公式には、実数解を表示する場合であっても、負の数の平方根が現れることがある。
- 整数の性質は、整数の概念を複素数にまで拡張したガウス整数を考えることで、より深く理解できる。
- このあたりの例を見ると、どうやら虚数って、整数とか実数とかいう対象を深く知ろうと思ったら必然的に出くわすもののようだ、という感覚になりませんか？
- 「ないけど便利だから人間が勝手に作った」というよりは、もっと本質的に、整数や実数と不可分なものっぽくない？

この章はこれで終わり

質問などあればいつでもどうぞ！

プログラミングの話

みなさんに質問

interface というプログラミング用語、知ってますか？

- Java, C#, TypeScript などでは普通に使われる概念だと思います。
- 実装ではなく interface に依存しましょう、とかよく言われるやつです。
- みんな知ってそうであれば説明は省略します。
- 知らなそうな人がいれば説明します。

interface

- interface (インターフェイス) とは、一言でいえば、実装のないクラスです。
- クラスなのでメソッドを持ちますが、interface のメソッドでは名前と型だけが定義されます。
- ある interface を実装 (継承) するクラスは、その interface の要請を満たす必要があります。
 - というのはやや曖昧な言い方で、もうちょっとプログラミング的な説明をすれば、interface で定義されたメソッドを実装する必要があります。
- ある interface を実装したクラスのインスタンスは、その interface 型の変数として扱えます。
 - たとえば、StringValidator インターフェイスを実装する ZipCodeValidator クラスと PhoneNumberValidator クラスがある場合、これらのクラスのインスタンスは StringValidator 型として統一的に扱うことができます。

interface と自然数

- 現代の数学では、何かを定義するときに interface のような形式をとることが多いです。
- 文字列を受け取ってブール値を返す `isAcceptable()` メソッドが定義されていれば `StringValidator` と呼びます、というのと同じような定義がしばしばなされます。

interface と自然数

- 現代の数学では、何かを定義するときに interface のような形式をとることが多いです。
- 文字列を受け取ってブール値を返す `isAcceptable()` メソッドが定義されていれば `StringValidator` と呼びます、というのと同じような定義がしばしばなされます。
- 「このような条件を満たす関数を C^1 級関数と呼びます」とか、
- 「このような条件を満たす位相空間を Hausdorff 空間と呼びます」とか。

interface と自然数

- 現代の数学では、何かを定義するときに interface のような形式をとることが多いです。
- 文字列を受け取ってブール値を返す `isAcceptable()` メソッドが定義されていれば `StringValidator` と呼びます、というのと同じような定義がしばしばなされます。
- 「このような条件を満たす関数を C^1 級関数と呼びます」とか、
- 「このような条件を満たす位相空間を Hausdorff 空間と呼びます」とか。
- 現代的には、自然数も interface のような形式で定義されます。
- その条件は、ペアノの公理と呼ばれるものです。

ペアノの公理

- ペアノの公理とは、次のようなものです。
- 「うわっ」と思ったら、無理に読まなくても大丈夫です。

ペアノの公理

集合 N, S, O が以下の条件を満たすとき、組 (N, S, O) をペアノシステムといい、 N の要素を自然数という。

- ① $O \in N$
- ② S は N から N への写像である
- ③ $\forall n \in N, S(n) \neq O$
- ④ $\forall m, n \in N, (S(m) = S(n) \Rightarrow m = n)$
- ⑤ $\forall A \subset N, ((O \in A \wedge (\forall n \in A, S(n) \in A)) \Rightarrow A = N)$

ペアノの公理をざっくり言い直すと

やや不正確になる可能性もありますが...

ペアノの公理 (ざっくり)

集合 N 、「次の数」を決めるルール S 、「最小の自然数」 0 が以下の条件を満たすとき、組 $(N, S, 0)$ をペアノシステムといい、 N の要素を自然数という。

- ① 0 は N の要素である。
 - ② N の要素 x に対して、「その次の数」 $S(x)$ もまた N の要素である。
 - ③ 0 は N のどの要素の「次の数」にもならない。
 - ④ N の要素 x, y に対して、それぞれの「次の数」 $S(x), S(y)$ が等しくなるのは、 x と y が等しい場合のみである。
 - ⑤ N は、上記の条件を満たすための最低限の要素しか含まない。
- interface のように、何かの条件を満たしさえすればなんでも自然数と呼ぶということだけ認識してくれば大丈夫です。

実数の公理

実数にも同じような公理があります。

実数の公理（ざっくり）

集合 R 、和 $+$ 、積 \times 、順序 \leq が以下の条件を満たすとき、 R の要素を実数という。

- $(R, +, \times, \leq)$ は順序体である。
- (R, \leq) の上に有界な部分集合は上限を持つ。
- もはやある程度の知識を前提とした言い方になっちゃってますが...
- やはり、interface のように何かを満たしさえすればなんでも実数と呼ぶということだけ認識してくれば大丈夫です。

自然数は存在するか？

いま、次の質問について考えましょう。

質問

自然数は存在するか？

自然数は存在するか？

いま、次の質問について考えましょう。

質問

自然数は存在するか？

これは、現代的には以下の質問と同じことです。

質問の言い換え

ペアノシステムの実装は存在するか？

自然数は存在するか？

いま、次の質問について考えましょう。

質問

自然数は存在するか？

これは、現代的には以下の質問と同じことです。

質問の言い換え

ペアノシステムの実装は存在するか？

問題がここまで明確になると、回答もしやすいです。

回答

存在する。

いろいろな実装が可能ですが、フォン・ノイマンの構成法が有名です。
興味があれば調べてみてください。

自然数の足し算と掛け算

- ペアノシステムでは、どの自然数にも「次の数」があります。
- これを使って、足し算と掛け算も「作る」ことができます。
- たとえば、 $n + 1$ は n の次の数のことです。
- $n + m$ は、 n に対して、次の数をとる操作を m 回繰り返したものです。
 - 自然数を作ろうとしているときに「 m 回繰り返す」ということを気軽に言うてはいけないので、本当は同様の操作をうまく表現します。
- $n \times m$ も、そんな感じで作れます。

交換則、結合則、分配則

このようにして作った足し算と掛け算には、以下の特徴があります。

交換則

$$a + b = b + a$$

$$ab = ba$$

結合則

$$(a + b) + c = a + (b + c)$$

$$(ab)c = a(bc)$$

分配則

$$a(b + c) = ab + ac$$

自然数から整数を作る

- ここからは、自然数は普通に知っているものとしましょう。
- フォン・ノイマンの方法で構成したペアノシステム、とか考えると無駄にわかりづらくなります。
- プログラミングにおいて実装を意識せずにインターフェイスのみを意識するのと同様に、どのように構成されるかよりも、どんな特徴を持っていて、どんな操作が可能なのかを気にするほうが重要です。
- いまから、自然数を使って整数を作ります。
 - 数字とか記号がたくさん出てくるので、「うわ無理」と思ったらすべて理解しようとしなくても大丈夫です。
 - 部分的にでも、なんとなく整数作ろうとしてるっぽいな、と思いながら見てください。

自然数の引き算

- 自然数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a + d = b + c$ で定義します。
 - 「差」 $a - b$ を、マイナス記号を使わずに定義しようとしています。
 - 例えば、 $(1, 4) = (2, 5) = (3, 6) = \dots$ です。
 - マイナス記号を使って表現すれば、 $1 - 4 = 2 - 5 = 3 - 6 = \dots$ です。

自然数の引き算

- 自然数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a + d = b + c$ で定義します。
 - 「差」 $a - b$ を、マイナス記号を使わずに定義しようとしています。
 - 例えば、 $(1, 4) = (2, 5) = (3, 6) = \dots$ です。
 - マイナス記号を使って表現すれば、 $1 - 4 = 2 - 5 = 3 - 6 = \dots$ です。
- 足し算 $(a, b) + (c, d)$ を $(a + c, b + d)$ で定義します。
 - $(a - b) + (c - d) = (a + c) - (b + d)$ のことです。

自然数の引き算

- 自然数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a + d = b + c$ で定義します。
 - 「差」 $a - b$ を、マイナス記号を使わずに定義しようとしています。
 - 例えば、 $(1, 4) = (2, 5) = (3, 6) = \dots$ です。
 - マイナス記号を使って表現すれば、 $1 - 4 = 2 - 5 = 3 - 6 = \dots$ です。
- 足し算 $(a, b) + (c, d)$ を $(a + c, b + d)$ で定義します。
 - $(a - b) + (c - d) = (a + c) - (b + d)$ のことです。
- 掛け算 $(a, b) \times (c, d)$ を $(ac + bd, ad + bc)$ で定義します。
 - $(a - b) \times (c - d) = (ac + bd) - (ad + bc)$ のことです。

自然数の引き算

- 自然数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a + d = b + c$ で定義します。
 - 「差」 $a - b$ を、マイナス記号を使わずに定義しようとしています。
 - 例えば、 $(1, 4) = (2, 5) = (3, 6) = \dots$ です。
 - マイナス記号を使って表現すれば、 $1 - 4 = 2 - 5 = 3 - 6 = \dots$ です。
- 足し算 $(a, b) + (c, d)$ を $(a + c, b + d)$ で定義します。
 - $(a - b) + (c - d) = (a + c) - (b + d)$ のことです。
- 掛け算 $(a, b) \times (c, d)$ を $(ac + bd, ad + bc)$ で定義します。
 - $(a - b) \times (c - d) = (ac + bd) - (ad + bc)$ のことです。
- この足し算と掛け算は、自然数のそれらと同じように、交換則、結合則、分配則を満たします。
 - ここでは証明はしませんが、少々複雑なだけで、難しくはないです。

自然数から整数を作る

- ここで、ペア $(n, 0)$ は自然数 n と自然に同一視できます。
- $(a, b) = (a, 0) + (0, 1) \times (b, 0)$ なので、 $(0, 1)$ を -1 とも書くことにすれば、上の同一視も含めて $(a, b) = a + (-1) \times b$ です。
- (a, b) を単に $a - b$ とも、 $(0, b)$ を単に $-b$ とも書くことにします。
- このようにして、自然数の引き算によって得られる数すべてと、その足し算、掛け算を作ることができました。
- この構造を整数と呼びます。
- これで、自然数から整数を作ることができました。

自然数から整数が作れた

- 自然数を使って引き算を「作る」ことができ、整数を「作る」ことができました。
- 1 の存在を認めるのであれば、 -1 の存在も認めるべきだと言えます。
 - そもそも今の文脈では 1 も正体不明でした (interface 的に定義されていました) が、1 が存在することと同じくらい確かに -1 が存在しているとは言えると思います。1 から -1 を作れるからです。

整数から複素数を作る

- ここからは、整数は普通に知っているものとしましょう。
- 自然数のペアに特殊な「等しさ」を導入したもの、とか考えると無駄にわかりづらくなります。
- プログラミングにおいて実装を意識せずにインターフェイスのみを意識するのと同様に、どのように構成されるかよりも、どんな操作が可能なのを気にするべきです。
- いまから、整数を使って複素数を作ります。
 - やはり数字とか記号がたくさん出てくるので、無理にすべて理解しようとしなくても大丈夫です。
 - 部分的にでも、なんとなく複素数作ろうとしてるっぽいな、と思いながら見てください。

整数から複素数を作る

- 整数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a = c$ かつ $b = d$ で定義します。
 - 「複素数」 $a + bi$ を、 i を使わずに定義しようとしています。

整数から複素数を作る

- 整数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a = c$ かつ $b = d$ で定義します。
 - 「複素数」 $a + bi$ を、 i を使わずに定義しようとしています。
- 足し算 $(a, b) + (c, d)$ を $(a + c, b + d)$ で定義します。

整数から複素数を作る

- 整数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a = c$ かつ $b = d$ で定義します。
 - 「複素数」 $a + bi$ を、 i を使わずに定義しようとしています。
- 足し算 $(a, b) + (c, d)$ を $(a + c, b + d)$ で定義します。
- 掛け算 $(a, b) \times (c, d)$ を $(ac - bd, ad + bc)$ で定義します。

整数から複素数を作る

- 整数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a = c$ かつ $b = d$ で定義します。
 - 「複素数」 $a + bi$ を、 i を使わずに定義しようとしています。
- 足し算 $(a, b) + (c, d)$ を $(a + c, b + d)$ で定義します。
- 掛け算 $(a, b) \times (c, d)$ を $(ac - bd, ad + bc)$ で定義します。
- この足し算と掛け算は、自然数や整数のそれらと同じように、交換則、結合則、分配則を満たします。

整数から複素数を作る

- 整数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a = c$ かつ $b = d$ で定義します。
 - 「複素数」 $a + bi$ を、 i を使わずに定義しようとしています。
- 足し算 $(a, b) + (c, d)$ を $(a + c, b + d)$ で定義します。
- 掛け算 $(a, b) \times (c, d)$ を $(ac - bd, ad + bc)$ で定義します。
- この足し算と掛け算は、自然数や整数のそれらと同じように、交換則、結合則、分配則を満たします。
- ここで、 $(a, 0)$ は整数 a と自然に同一視できます。
- $(a, b) = (a, 0) + (b, 0) \times (0, 1)$ なので、 $(0, 1)$ を i とも書き、 (a, b) を $a + bi$ とも書くことにします。

整数から複素数を作る

- 整数のペア (a, b) を考えます。
- 「等しさ」 $(a, b) = (c, d)$ を $a = c$ かつ $b = d$ で定義します。
 - 「複素数」 $a + bi$ を、 i を使わずに定義しようとしています。
- 足し算 $(a, b) + (c, d)$ を $(a + c, b + d)$ で定義します。
- 掛け算 $(a, b) \times (c, d)$ を $(ac - bd, ad + bc)$ で定義します。
- この足し算と掛け算は、自然数や整数のそれらと同じように、交換則、結合則、分配則を満たします。
- ここで、 $(a, 0)$ は整数 a と自然に同一視できます。
- $(a, b) = (a, 0) + (b, 0) \times (0, 1)$ なので、 $(0, 1)$ を i とも書き、 (a, b) を $a + bi$ とも書くことにします。
- ここで、 $i = \sqrt{-1}$ です。
 - $i^2 = (0, 1) \times (0, 1) = (-1, 0) = -1$ だからです。
- この整数のペアと、「等しさ」と、足し算と、掛け算の構造を複素数と呼びます。
- これで、整数から複素数を作ることができました。

整数から複素数が作れた

- 整数を使って複素数を「作る」ことができました。
- -1 の存在を認めるのであれば、 $\sqrt{-1}$ の存在も認めるべきだと言えます。
 - そもそも今の文脈では -1 も正体不明でした（やや機械的な操作で無理やり作ったような印象がありました）が、 -1 が存在することと同じくらい確かに $\sqrt{-1}$ が存在しているとは言えると思います。 -1 から $\sqrt{-1}$ を作れるからです。

いったんまとめ

- 現代の数学においては、interface 的な定義がなされることがけっこう多い。
- 自然数や実数もそうで、どのように作られるかに関係なく、一定の条件を満たしてさえいれば自然数や実数と呼ばれる。
- その意味で、自然数は存在する（ペアノシステムは実装可能）。
- 自然数があれば、整数が作れる。
- 整数があれば、複素数（虚数）が作れる。

この章はこれで終わり

質問などあればいつでもどうぞ！

あらためて、最初の質問

質問 1

$\sqrt{-1}$ は存在するか？

質問 2

-1 は存在するか？

質問 3

1 は存在するか？

- ここまでの話をもとに、みなさんはどう思われますか？

あらためて、もう1つの質問

質問

数とは何か？

- ここまでの話をもとに、みなさんはどう思われますか？

最後に、筆者の感覚

筆者の感覚では、

- 自然数は存在します。
- 1 や 3 や 7 といった数は物理的には「ない」し、観測することも不可能な概念だと思いますが、それでも確かに存在しているものと考えてよいと思います。
 - ある意味、「幽霊は存在する」と同じようなノリで言ってます。

最後に、筆者の感覚

筆者の感覚では、

- **自然数は存在します。**
- 1 や 3 や 7 といった数は物理的には「ない」し、観測することも不可能な概念だと思いますが、それでも確かに存在しているものと考えてよいと思います。
 - ある意味、「幽霊は存在する」と同じようなノリで言ってます。
- 同じように、**負の数は存在します。**
- やはり物理的には「ない」ですが、確かに存在しているものと考えてよいと思います。

最後に、筆者の感覚

筆者の感覚では、

- **自然数は存在します。**
- 1 や 3 や 7 といった数は物理的には「ない」し、観測することも不可能な概念だと思いますが、それでも確かに存在しているものと考えてよいと思います。
 - ある意味、「幽霊は存在する」と同じようなノリで言ってます。
- 同じように、**負の数は存在します。**
- やはり物理的には「ない」ですが、確かに存在しているものと考えてよいと思います。
- 同じように、**虚数は存在します。**
- やはり物理的には「ない」ですが、確かに存在しているものと考えてよいと思います。

最後に、筆者の感覚

筆者の感覚では、

- **自然数は存在します。**
- 1 や 3 や 7 といった数は物理的には「ない」し、観測することも不可能な概念だと思いますが、それでも確かに存在しているものと考えてよいと思います。
 - ある意味、「幽霊は存在する」と同じようなノリで言ってます。
- 同じように、**負の数は存在します。**
- やはり物理的には「ない」ですが、確かに存在しているものと考えてよいと思います。
- 同じように、**虚数は存在します。**
- やはり物理的には「ない」ですが、確かに存在しているものと考えてよいと思います。
- それぞれの存在の確かさは、筆者にとってまったく同じです。
- たまに聞く「自然数は存在するけど虚数は存在しない」という主張は、どういう意味で言ってるんだろーな、と不思議に思います。

おわり

- おわりです。
- Thank you for your attention!