

君は cmath を知っているか

Hayao Suzuki

PyCon mini Shizuoka 2020

February 29, 2020

# Contents

- ① 自己紹介
- ② cmath とは何者か
- ③ 複素数とは何か
- ④ 複素数の極座標表記

# 自己紹介

## お前誰よ

名前 Hayao Suzuki (鈴木 駿)

Twitter @CardinalXaro

ブログ <https://xaro.hatenablog.jp/>

専門 数学 (組合せ論・グラフ理論)

学位 修士 (工学)、電気通信大学

仕事 酢豆腐スペシャリスト

## 技術書の査読

- 『Effective Python』 (オライリージャパン)
- 『エレガントな SciPy』 (オライリージャパン)
- 『データサイエンス設計マニュアル』 (オライリージャパン) など
- <https://xaro.hatenablog.jp/> に一覧あります。

## いろんな発表

- 「SymPy による数式処理」 (PyCon JP 2018)
- 「Python で楽しむ初等整数論」 (PyCon mini Hiroshima 2019) など
- <https://xaro.hatenablog.jp/> に一覧あります。

## バッテリー同梱哲学 (PEP 206 より)

Python ディストリビューション自身が、別途ダウンロードすることなくすぐに利用できる豊富で汎用性の高い標準ライブラリを持つこと。

## Python チュートリアルで紹介されている例

- `xmlrpc.client` XML-RPC クライアント
- `xmlrpc.server` XML-RPC サーバー
- `email` 電子メールと MIME 処理のためのパッケージ
- `json` JSON エンコーダおよびデコーダ
- `sqlite3` SQLite データベースに対する DB-API 2.0 インタフェース

## cmath とは何者か

- ① C 言語で実装された高速な math ライブラリ
- ② キュウリ (Cucumber) の画像識別のための数学ライブラリ
- ③ 複素数 (Complex Number) の計算ライブラリ

## cmath モジュール

- 複素数のための数学関数
- 9V 電池やらニカド電池のような存在に今、スポットを当てる。

## 今回使うもの

- Python 3.7.x
- Matplotlib (グラフ描画ライブラリ)
- SymPy (記号計算ライブラリ、もちろん複素数も扱える)

## 君は `cmath` を知っているか

- 複素数とは何か
- 複素数の極座標表記
- 複素指数関数・三角関数
- Mandelbrot 集合

## 資料は設計図共有サイトにある！

資料はすべて

<https://github.com/HayaoSuzuki/PyCon-mini-Shizuoka-2020/> にあります。



複素数の定義を言えますか？

# 複素数の定義

## 定義（複素数）

$i^2 = -1$  であるような基底が  $1, i$  を持つ実数体  $\mathbf{R}$  上の 2 次元ベクトル空間の元を複素数と呼ぶ。また、 $i$  を虚数単位と呼ぶ。

## Python で複素数を定義する

```
>>> 3 + 5j # Python では虚数単位を j または J とする
(3+5j)
>>> (0 + 1j)**2 # 虚数単位の自乗は-1 となる。
(-1+0j)
>>> 4 + 5j == (5j + 4) # 実部と虚部がそれぞれ等しい
True
```

体 (Field) == 四則演算ができる集合

複素数は複素数体  $\mathbb{C}$  をなす。

## Python における複素数の四則演算

```
>>> 8 - 5j + -5 + 1j # 加法  
(3-4j)  
>>> (1 + 2j) * (1 - 2j) # 乗法  
(5+0j)  
>>> (97 + 0j) / (4 + 9j) # 除法  
(3.9999999999999996-9j)
```

複素数体は順序体ではない

実数のような全順序関係を定義できない！

Python も複素数体は順序体ではないことを知っている

```
>>> -100 - 100j < 65536 + 256j # 右辺が大きそうに思えるが...
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
TypeError: '<' not supported between  
instances of 'complex' and 'complex'
```

## 複素数平面

複素数  $z = x + iy$  を 2 次元実数平面  $\mathbf{R}^2$  上の点  $(x, y)$  とみなすことができる。これを複素数平面という。

## 複素数の極座標形式

複素数平面上の点  $z = x + iy (x, y \in \mathbf{R})$  を実部  $x$  と虚部  $y$  の組  $(x, y)$  ではなく原点からの距離  $r$  と偏角  $\theta$  の組  $(r, \theta)$  でも定義できる。これを複素数の極座標形式という。

# 複素数の極座標表記

百聞は一見に然り

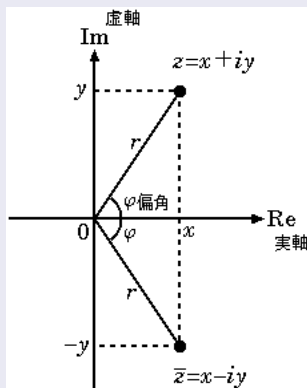


Figure: 複素数平面 (Wikipedia から引用)

# Python における複素数の極座標表記

## Python における複素数の四則演算

```
>>> import cmath # 真打登場
>>> z = 1 + 2j # 直交座標から極座標に変換する
>>> r, phi = cmath.polar(z)
>>> r, phi # r = abs(z), phi = cmath.phase(z)
(2.23606797749979, 1.1071487177940904)
>>> w = cmath.rect(r, phi) # 極座標から直交座標に変換する
>>> w
(1.0000000000000002+2j)
>>> cmath.isclose(z, w) # == ではなく isclose を使う
True
```