Complex math

All Versions

Examples

Advanced complex arithmetic

The module `cmath` includes additional functions to use complex numbers.

```
import cmath
```

This module can calculate the phase of a complex number, in radians:

```
z = 2+3j # A complex number
cmath.phase(z) # 0.982793723247329
```

It allows the conversion between the cartesian (rectangular) and polar representations of complex numbers:

```
cmath.polar(z) # (3.605551275463989, 0.982793723247329)
cmath.rect(2, cmath.pi/2) # (0+2j)
```

The module contains the complex version of

- Exponential and logarithmic functions (as usual, `log` is the natural logarithm and `log10` the decimal logarithm):

  ```
  cmath.exp(z) # (-7.315110094901103+1.0427436562359045j)
  cmath.log(z) # (1.2824746787307684+0.982793723247329j)
  cmath.log10(-100) # (2+1.3643763538418412j)
  ```

- Square roots:

  ```
  cmath.sqrt(z) # (1.6741492280355401+0.8959774761298381j)
  ```

- Trigonometric functions and their inverses:

  ```
  cmath.sin(z)  # (9.15449914691143-4.168906959966565j)
  cmath.cos(z)  # (-4.189625690968807-9.109227893755337j)
  cmath.tan(z)  # (-0.003764025641504249+1.003238627353361j)
  cmath.asin(z) # (0.5706527843210994+1.9833870299165355j)
  cmath.acos(z) # (1.0001435424737972-1.9833870299165355j)
  cmath.atan(z) # (1.4099210495965755+0.22907268296853878j)
  cmath.sin(z)**2 + cmath.cos(z)**2 # (1+0j)
  ```

- Hyperbolic functions and their inverses:

  ```
  cmath.sinh(z)  # (-3.59056458998578+0.5309210862485197j)
  cmath.cosh(z)  # (-3.7245455049153224+0.5118225699873846j)
  cmath.tanh(z)  # (0.965385879022133-0.009884375038322495j)
  cmath.asinh(z) # (0.5706527843210994+1.9833870299165355j)
  cmath.acosh(z) # (1.9833870299165355+1.0001435424737972j)
  cmath.atanh(z) # (0.14694666622552977+1.3389725222944935j)
  cmath.cosh(z)**2 - cmath.sin(z)**2  # (1+0j)
  cmath.cosh((0+1j)*z) - cmath.cos(z) # 0j
  ```

Basic complex arithmetic

Python has built-in support for complex arithmetic. The imaginary unit is denoted by `j` :

```
z = 2+3j # A complex number
w = 1-7j # Another complex number
```

Complex numbers can be summed, subtracted, multiplied, divided and exponentiated:

```
z + w # (3-4j)
z - w # (1+10j)
z * w # (23-11j)
z / w # (-0.38+0.34j)
z**3  # (-46+9j)
```

Python can also extract the real and imaginary parts of complex numbers, and calculate their absolute value and conjugate:

```
z.real # 2.0
z.imag # 3.0
abs(z) # 3.605551275463989
z.conjugate() # (2-3j)
```

Syntax

```
cmath.rect(AbsoluteValue, Phase)
```

Parameters

Remarks