

## 3.6 复数的数学运算¶

### 问题¶

你写的最新的网络认证方案代码遇到了一个难题，并且你唯一的解决办法就是使用复数空间。再或者是你仅仅需要使用复数来执行一些计算操作。

### 解决方案¶

复数可以用使用函数 `complex(real, imag)` 或者是带有后缀j的浮点数来指定。比如：

```
>>> a = complex(2, 4)
>>> b = 3 - 5j
>>> a
(2+4j)
>>> b
(3-5j)
>>>
```

对应的实部、虚部和共轭复数可以很容易的获取。就像下面这样：

```
>>> a.real
2.0
>>> a.imag
4.0
>>> a.conjugate()
(2-4j)
>>>
```

另外，所有常见的数学运算都可以工作：

```
>>> a + b
(5-1j)
>>> a * b
(26+2j)
>>> a / b
(-0.4117647058823529+0.6470588235294118j)
>>> abs(a)
4.47213595499958
>>>
```

如果要执行其他的复数函数比如正弦、余弦或平方根，使用 `cmath` 模块：

```
>>> import cmath
>>> cmath.sin(a)
(24.83130584894638-11.356612711218174j)
>>> cmath.cos(a)
(-11.36423470640106-24.814651485634187j)
>>> cmath.exp(a)
(-4.829809383269385-5.5920560936409816j)
>>>
```

### 讨论¶

Python中大部分与数学相关的模块都能处理复数。比如如果你使用 `numpy`，可以很容易的构造一个复数数组并在这个数组上执行各种操作：

```
>>> import numpy as np
>>> a = np.array([2+3j, 4+5j, 6-7j, 8+9j])
>>> a
array([ 2.+3.j,  4.+5.j,  6.-7.j,  8.+9.j])
>>> a + 2
array([ 4.+3.j,  6.+5.j,  8.-7.j, 10.+9.j])
>>> np.sin(a)
```

```
array([ 9.15449915 -4.16890696j, -56.16227422 -48.50245524j,  
       -153.20827755-526.47684926j, 4008.42651446-589.49948373j])  
>>>
```

Python的标准数学函数确实情况下并不能产生复数值，因此你的代码中不可能会出现复数返回值。比如：

```
>>> import math  
>>> math.sqrt(-1)  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
ValueError: math domain error  
>>>
```

如果你想生成一个复数返回结果，你必须显示的使用 `cmath` 模块，或者在某个支持复数的库中声明复数类型的使用。比如：

```
>>> import cmath  
>>> cmath.sqrt(-1)  
1j  
>>>
```