

Примеры

Пример 1

Ввод [20]:

```
x = complex(1, 3)
y = complex(2, -1)
z = x * y
print(z)
g = complex(1, -2)
print(g)
t = complex(10, 0)
print(t)
h = t/g
print(h)
p = complex(-1, -1)
n = p * p
print(n)
C = z + h + n
print(C)
```

```
(5+5j)
(1-2j)
(10+0j)
(2+4j)
2j
(7+11j)
```

Пример 2

Ввод [21]:

```
x = complex(0, 1)
y = pow(x, 2)
print(y)
```

```
(-1+0j)
```

Пример 3

Ввод [22]:

```
x = complex(1, 3)
y = complex(2, -1)
z = x * y
print(z)
g = complex(1, -2)
print(t)
h = t / g
print(h)
p = complex(-1, -1)
n = p * p
print(n)
C = z + h + n
print(C)
```

```
(5+5j)
(10+0j)
(2+4j)
2j
(7+11j)
```

Пример 4

Ввод [23]:

```
import math
from sympy import *
x = Symbol('x')
print(solve(x ** 2 - 2 * x + 5))
```

```
[1 - 2*I, 1 + 2*I]
```

Пример 5

Ввод [24]:

```
x = complex(1, -2)
i = complex(0, 1)
f = x ** 4 + (2 + i) / x - (-3 + 2 * i)
print(f)
```

```
(-4+23j)
```

Пример 6

Ввод [25]:

```
(1 + i) ** 8 / (1 + i) ** 6
```

Out[25]:

```
(-0+2j)
```

Пример 7

Ввод [26]:

```

from sympy import Symbol, nsolve
import sympy
import mpmath
mpmath.mp.dps = 3
x = Symbol('x')
y = Symbol('y')
i = complex(0, 1)
f1 = (2 + i) * x + y * (2 - i) - 6
f2 = (2 - i) * x + (3 - 2 * i) * y - 8
print(nsolve((f1, f2), (x, y), (-1, 1)))

```

Matrix([[-0.0588 - 0.765*I], [1.82 + 1.71*I]])

Пример 8

Ввод [27]:

```
solve(x ** 2 - 3 + 4 * i)
```

Out[27]:

[-2.0 + 1.0*I, 2.0 - 1.0*I]

Пример 9

Ввод [28]:

```

x = Symbol('x')
i = complex(0, 1)
print(solve((2 + i) * x ** 2 - (5 - i) * x + 2 - 2 * i))

```

[0.8 - 0.4*I, 1.0 - 1.0*I]

Пример 10

Ввод [29]:

```

x = Symbol('x')
i = complex(0, 1)
print(solve(x ** 2 - 3 + 4 * i))

```

[-2.0 + 1.0*I, 2.0 - 1.0*I]

Пример 11

Ввод [30]:

```

i = complex(0, 1)
-(3 + 5 * i) ** 10 - 25 * (3 * i - 9) / (2 + 8 * i)

```

Out[30]:

(28984573.79411765+34989571.323529415j)

Пример 12

Ввод [33]:

```
import cmath
import math

abs(z)
```

Out[33]:

3.9999999999999996

Ввод [34]:

```
z = complex(2, 2 * sqrt(3))
cmath.phase(z)
round(math.degrees(cmath.phase(z)))
```

Out[34]:

60

Пример 13

Ввод [35]:

```
z1 = complex(-4, -9)
z2 = complex(1, -8)
complex(z1 - conjugate(z2)) / complex(z2 + conjugate(z1))
```

Out[35]:

 $(-0.19999999999999982 + 5.6000000000000005j)$

Ввод [36]:

```
i = complex(0, 1)
print((1 + 2 * i) * (-1 + 5 * i) / (6 - i))
```

 $(-1.8648648648648647 + 0.1891891891891892j)$

Ввод [37]:

```
z = complex(1, 2)
p = (1 + 3 * i) * z ** 2 + (-5 + 6 * i) * z + (2 - i)
print(p)
```

 $(-30 - 10j)$

Задача для самостоятельного решения

Вычислить модуль и аргумент числа $z = -8i$. Ответ: $|z| = 8, \arg(z) = -\frac{\pi}{2}$.

Ввод [44]:

```
z = complex(0, -8)
print (f'z = {z}')
print(abs(z))
print(cmath.phase(z))
```

```
z = -8j
8.0
-1.5707963267948966
```

Дополнительная задача

Создать объект комплексное число, который будет предоставлять математический функционал по работе с комплексными числами. Решить при помощи этого объекта задачу: Даны 2 комплексных числа $z = (-1, 2)$ и $w = (3, 2)$, рассчитать $5z - 3w$, построить график отображающий данное выражение.

Ввод [16]:

```
import matplotlib.pyplot as plt
```

Ввод [17]:

```
class ComplexNumber:
    def __init__(self, a, bj):
        self.num = complex(a, bj)

    def change_num(self, a, bj):
        self.num = complex(a, bj)

    def get_module(self):
        return abs(self.num )

    def sum_number(self, complex_num):
        return complex_num + self.num

    def subtract_number(self, complex_num):
        return self.num - complex_num

    def multiply_number(self, a):
        return a * self.num

    def division_number(self, a):
        return self.num / a
```

Ввод [18]:

```
z = ComplexNumber(-1, 2)
```

Ввод [19]:

```
w = ComplexNumber(3, 2)
```

Ввод [20]:

```
z_m = z.multiply_number(5)
```

Ввод [21]:

```
w_m = w.multiply_number(3)
```

Ввод [22]:

```
result = z_m - w_m
```

Ввод [23]:

```
print(z_m)  
print(w_m)  
print(result)
```

```
(-5+10j)
```

```
(9+6j)
```

```
(-14+4j)
```

Ввод [24]:

```
w_m.real, w_m.imag
```

Out[24]:

```
(9.0, 6.0)
```

Ввод [25]:

```
z_m.real, z_m.imag
```

Out[25]:

```
(-5.0, 10.0)
```

Ввод [26]:

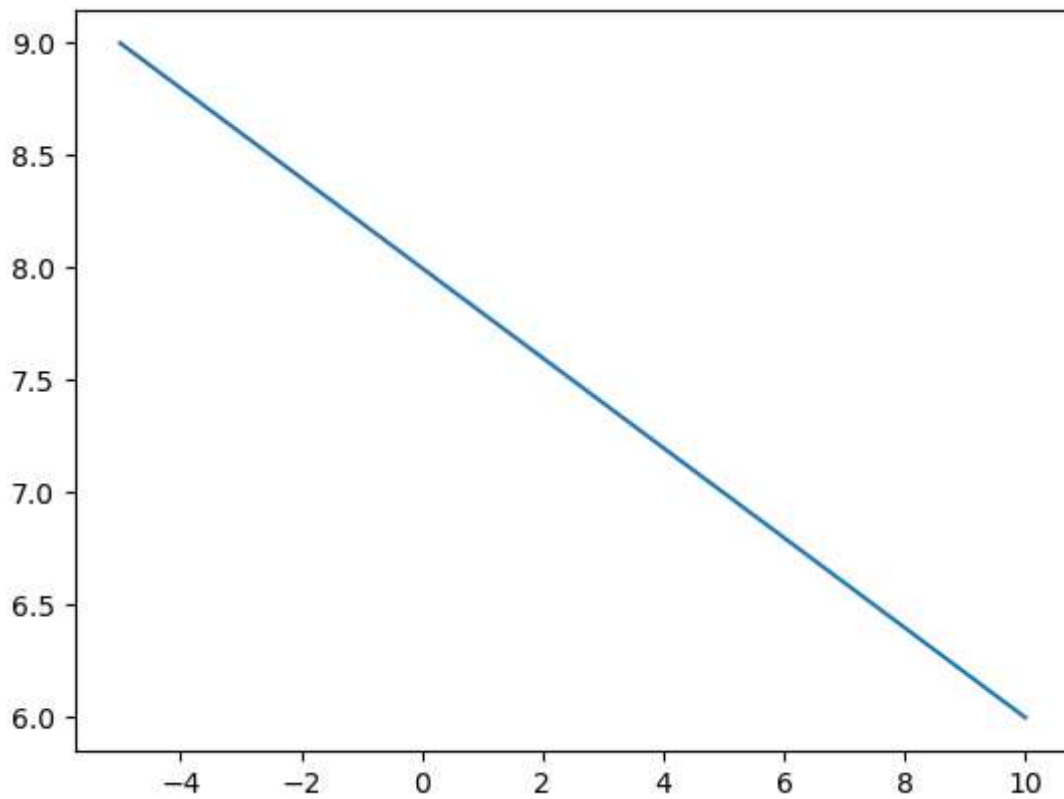
```
result.real, result.imag
```

Out[26]:

```
(-14.0, 4.0)
```

Ввод [33]:

```
plt.plot((z_m.real, z_m.imag), (w_m.real, w_m.imag), label='5z')  
plt.show()
```



Ввод [35]:

Out[35]:

-5

Ввод []: