

TÁC GIẢ: 8 SYNC

CỘNG ĐỒNG



[Kevin Nguyễn](#)



[Nhóm Chia Sẻ Công Nghệ](#)



[Nhóm BlockChain](#)



[Tiktok: 8 Sync](#)



[Youtube: 8 Sync Dev](#)



[Zalo](#)

KHÓA HỌC:



[Fullstack Python](#)



[Fullstack Nextjs](#)



[Fullstack Android-IOs](#)

Tài liệu sẽ được cập nhật định kì và thông báo trong group nên các bạn chú ý nhen .



PYTHON - CÁC TOÁN TỬ SỐ HỌC

Trong Python, các số là loại dữ liệu được sử dụng phổ biến nhất. Python sử dụng các ký hiệu giống như các phép toán số học cơ bản mà mọi người đều quen thuộc, tức là "+" cho phép cộng, "-" cho phép trừ, "*" cho phép nhân (hầu hết các ngôn ngữ

lập trình sử dụng "" thay vì "x" như được sử dụng trong toán học / đại số), và "/" cho phép chia (một lần nữa cho "÷" được sử dụng trong Toán học).

Ngoài ra, Python định nghĩa một số toán tử số học khác. Chúng là "%" (Phần dư), "**" (Lũy thừa) và "//" (Chia lấy phần nguyên).

CÁC TOÁN TỬ SỐ HỌC PYTHON

Các toán tử số học là các toán tử nhị phân vì chúng hoạt động trên hai toán hạng. Python hỗ trợ hoàn toàn phép tính hỗn hợp. Đó là, hai toán hạng có thể thuộc hai loại số khác nhau. Trong tình huống như vậy, Python mở rộng toán hạng hẹp hơn. Một đối tượng số nguyên hẹp hơn một đối tượng số thực, và số thực hẹp hơn số phức. Do đó, kết quả của phép toán số học giữa số nguyên và số thực là một số thực. Kết quả của số thực và một số phức là một số phức, tương tự, phép tính trên một số nguyên và một đối tượng số phức sẽ cho kết quả là một đối tượng số phức.

CÁC TOÁN TỬ SỐ HỌC KHÁC NHAU TRONG PYTHON

Dưới đây là bảng liệt kê tất cả các toán tử số học có sẵn trong Python:

Toán tử	Tên	Ví dụ
+	Cộng	$a + b = 30$
-	Trừ	$a - b = -10$
*	Nhân	$a * b = 200$
/	Chia	$b / a = 2$
%	Phần dư	$b \% a = 0$
**	Lũy thừa	$a^b = 1020$
//	Chia lấy dư	$9 // 2 = 4$

Hãy cùng tìm hiểu các toán tử này thông qua các ví dụ.

TOÁN TỬ CỘNG (+)

Toán tử này được phát âm là "cộng", là một toán tử số học cơ bản. Nó cộng hai toán hạng số học ở hai bên và trả về kết quả cộng.

VÍ DỤ: CỘNG HAI SỐ NGUYÊN

Trong ví dụ sau, hai biến số nguyên là các toán hạng cho toán tử "+".



< 8 Sync Dev />

```
a=10
b=20
print ("Tổng của hai số nguyên:")
print ("a =",a,"b =",b,"tổng =",a+b)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Tổng của hai số nguyên
a = 10 b = 20 tổng = 30
```

VÍ DỤ: CỘNG SỐ NGUYÊN VÀ SỐ THỰC

Cộng một số nguyên và một số thực sẽ cho kết quả là một số thực.



< 8 Sync Dev />

```
a=10
b=20.5
print ("Tổng của số nguyên và số thực")
print ("a =",a,"b =",b,"tổng =",a+b)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

Tổng của số nguyên và số thực

$a = 10$ $b = 20.5$ tổng = 30.5

VÍ DỤ: CỘNG HAI SỐ PHỨC

Kết quả của việc thêm số thực vào số phức là một số phức.



< 8 Sync Dev />

```
a=10+5j
```

```
b=20.5
```

```
print ("Tổng của số phức và số thực")
```

```
print ("a=",a,"b=",b,"tổng=",a+b)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

Tổng của số phức và số thực

$a = (10+5j)$ $b = 20.5$ tổng = $(30.5+5j)$

TOÁN TỬ TRỪ (-)

Toán tử này, được biết đến như là "trừ", trừ toán hạng thứ hai khỏi toán hạng đầu tiên. Số kết quả là âm nếu toán hạng thứ hai lớn hơn.

VÍ DỤ: TRỪ HAI SỐ NGUYÊN

Ví dụ đầu tiên cho thấy sự trừ của hai số nguyên.



< 8 Sync Dev />

```
a=10
b=20
print ("Hiệu của hai số nguyên:")
print ("a =",a,"b =",b,"a-b =",a-b)
print ("a =",a,"b =",b,"b-a =",b-a)
```

Kết quả –



< 8 Sync Dev />

```
Hiệu của hai số nguyên
a = 10 b = 20 a-b = -10
a = 10 b = 20 b-a = 10
```

VÍ DỤ: TRỪ SỐ NGUYÊN VÀ SỐ THỰC

Phép trừ của một số nguyên và một số thực tuân theo nguyên tắc tương tự.



< 8 Sync Dev />

```
a=10
b=20.5
print ("Hiệu của python số nguyên và số thực")
print ("a =",a,"b =",b,"a-b =",a-b)
print ("a =",a,"b =",b,"b-a =",b-a)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

Hiệu của số nguyên và số thực

```
a= 10 b= 20.5 a-b= -10.5
```

```
a= 10 b= 20.5 b-a= 10.5
```

VÍ DỤ: TRỪ HAI SỐ PHỨC

Trong phép trừ liên quan đến một số phức và một số thực, phần thực được sử dụng trong phép tính.



< 8 Sync Dev />

```
a=10+5j
```

```
b=20.5
```

```
print ("Hiệu của số phức và số thực")
```

```
print ("a=",a,"b=",b,"a-b=",a-b)
```

```
print ("a=",a,"b=",b,"b-a=",b-a)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

Hiệu của số phức và số thực

```
a= (10+5j) b= 20.5 a-b= (-10.5+5j)
```

```
a= (10+5j) b= 20.5 b-a= (10.5-5j)
```

TOÁN TỬ NHÂN (*)

Ký hiệu * (dấu hoa thị) được định nghĩa như một toán tử nhân trong Python (như trong nhiều ngôn ngữ). Nó trả về tích của hai toán hạng ở hai bên của nó. Nếu bất

kỳ toán hạng nào là số âm, kết quả cũng là số âm. Nếu cả hai đều âm, kết quả là dương. Thay đổi thứ tự của các toán hạng không làm thay đổi kết quả.

VÍ DỤ: NHÂN HAI SỐ NGUYÊN



< 8 Sync Dev />

```
a=10
b=20
print ("Tích của hai số nguyên")
print ("a =",a,"b =",b,"a*b =",a*b)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Tích của hai số nguyên
a = 10 b = 20 a*b = 200
```

VÍ DỤ: NHÂN SỐ NGUYÊN VÀ SỐ THỰC

Trong phép nhân, một toán hạng số thực có thể có một biểu thức thập phân tiêu chuẩn, hoặc một biểu thức khoa học.



< 8 Sync Dev />

```
a=10
b=20.5
print ("Tích của số nguyên và số thực")
print ("a =",a,"b =",b,"a*b =",a*b)

a=-5.55
```

```
b=6.75E-3
```

```
print ("Tích của số thực và số thực")
```

```
print ("a =",a,"b =",b,"a*b =",a*b)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

Tích của số nguyên và số thực

a = 10 b = 20.5 a-b = -10.5

Tích của số thực và số thực

a = -5.55 b = 0.00675 a*b = -0.037462499999999996

VÍ DỤ: NHÂN VỚI SỐ PHỨC

Đối với phép nhân có một toán hạng số phức, toán hạng khác nhân cả phần thực và phần ảo.



< 8 Sync Dev />

```
a=10+5j
```

```
b=20.5
```

```
print ("Tích của số phức và số thực")
```

```
print ("a =",a,"b =",b,"a*b =",a*b)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

Tích của số phức và số thực

a = (10+5j) b = 20.5 a*b = (205+102.5j)

TOÁN TỬ CHIA (/)

Toán tử "/" (dấu gạch chéo) thường được gọi là gạch chéo phía trước. Kết quả của toán tử chia là tử số (toán hạng bên trái) chia cho mẫu số (toán hạng bên phải). Số kết quả là âm nếu bất kỳ toán hạng nào là số âm. Vì vô cùng không thể được lưu trữ trong bộ nhớ, Python ném ra lỗi `ZeroDivisionError` nếu mẫu số là 0. Kết quả của toán tử chia trong Python luôn là một số thực, ngay cả khi cả hai toán hạng đều là số nguyên.

VÍ DỤ: CHIA HAI SỐ NGUYÊN



< 8 Sync Dev />

```
a=10
b=20
print ("Chia của hai số nguyên")
print ("a=",a,"b=",b,"a/b=",a/b)
print ("a=",a,"b=",b,"b/a=",b/a)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Chia của hai số nguyên
a= 10 b= 20 a/b= 0.5
a= 10 b= 20 b/a= 2.0
```

VÍ DỤ: CHIA VỚI SỐ THỰC

Trong phép chia, một toán hạng số thực có thể có một biểu thức thập phân tiêu chuẩn, hoặc một biểu thức khoa học.



< 8 Sync Dev />

```
a=10
b=-20.5
print ("Chia của số nguyên và số thực")
print ("a=",a,"b=",b,"a/b=",a/b)

a=-2.50
b=1.25E2
print ("Chia của số thực và số thực")
print ("a=",a,"b=",b,"a/b=",a/b)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Chia của số nguyên và số thực
a= 10 b= -20.5 a/b= -0.4878048780487805
Chia của số thực và số thực
SỐ nguyên và số thực
```python
a=10
b=20.5
print ("Hiệu của số nguyên và số thực")
print ("a=",a,"b=",b,"a-b=",a-b)
print ("a=",a,"b=",b,"b-a=",b-a)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Hiệu của số nguyên và số thực
a= 10 b= 20.5 a-b= -10.5
a= 10 b= 20.5 b-a= 10.5
```

## VÍ DỤ: TRỪ HAI SỐ PHỨC

Trong phép trừ liên quan đến một số phức và một số thực, thành phần thực được sử dụng trong phép tính.



< 8 Sync Dev />

```
a=10+5j
b=20.5
print ("Hiệu của số phức và số thực")
print ("a=",a,"b=",b,"a-b=",a-b)
print ("a=",a,"b=",b,"b-a=",b-a)
```

Nó sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Hiệu của số phức và số thực
a= (10+5j) b= 20.5 a-b= (-10.5+5j)
a= (10+5j) b= 20.5 b-a= (10.5-5j)
```

## TOÁN TỬ NHÂN (\*)

Ký hiệu \* (asterisk) được định nghĩa là toán tử nhân trong Python (như trong nhiều ngôn ngữ). Nó trả về tích của hai toán hạng ở hai bên. Nếu bất kỳ toán hạng nào là số âm, kết quả cũng là số âm. Nếu cả hai đều âm, kết quả là dương. Thay đổi thứ tự của các toán hạng không làm thay đổi kết quả.

## VÍ DỤ: NHÂN HAI SỐ NGUYÊN



< 8 Sync Dev />



```
a=10
b=20
print ("Tích của hai số nguyên")
print ("a =",a,"b =",b,"a*b =",a*b)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Tích của hai số nguyên
a = 10 b = 20 a*b = 200
```

## VÍ DỤ: NHÂN SỐ NGUYÊN VÀ SỐ THỰC

Trong phép nhân, một toán hạng số thực có thể có một biểu diễn thập phân tiêu chuẩn hoặc một biểu diễn toán học.



< 8 Sync Dev />

```
a=10
b=20.5
print ("Tích của số nguyên và số thực")
print ("a =",a,"b =",b,"a*b =",a*b)

a=-5.55
b=6.75E-3
print ("Tích của số thực và số thực")
print ("a =",a,"b =",b,"a*b =",a*b)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

Tích của số nguyên và số thực

$a = 10$   $b = 20.5$   $a - b = -10.5$

Tích của số thực và số thực

$a = -5.55$   $b = 0.00675$   $a * b = -0.037462499999999996$

## VÍ DỤ: NHÂN VỚI SỐ PHỨC

Đối với phép nhân liên quan đến một toán hạng phức, toán hạng còn lại nhân cả phần thực và phần ảo.



< 8 Sync Dev />

```
a=10+5j
b=20.5
print ("Tích của số phức và số thực")
print ("a =",a,"b =",b,"a*b =",a*b)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Tích của số phức và số thực
a = (10+5j) b = 20.5 a*b = (205+102.5j)
```

## TOÁN TỬ CHIA (/)

Dấu "/" thường được gọi là dấu gạch chéo. Kết quả của toán tử chia là tử số (toán hạng trái) chia cho mẫu số (toán hạng phải). Số kết quả là âm nếu bất kỳ toán hạng nào là số âm. Vì vô hạn không thể được lưu trữ trong bộ nhớ, Python ném ra `ZeroDivisionError` nếu mẫu số là 0. Kết quả của toán tử chia trong Python luôn là một số thực, ngay cả khi cả hai toán hạng đều là số nguyên.

# VÍ DỤ: CHIA HAI SỐ NGUYÊN



< 8 Sync Dev />

```
a=10
b=20
print ("Chia của hai số nguyên")
print ("a=",a,"b=",b,"a/b=",a/b)
print ("a=",a,"b=",b,"b/a=",b/a)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Chia của hai số nguyên
a= 10 b= 20 a/b= 0.5
a= 10 b= 20 b/a= 2.0
```

# VÍ DỤ: CHIA VỚI SỐ THỰC

Trong phép chia, một toán hạng số thực có thể có một biểu diễn thập phân tiêu chuẩn hoặc một biểu diễn toán học.



< 8 Sync Dev />

```
a=10
b=-20.5
print ("Chia của số nguyên và số thực")
print ("a=",a,"b=",b,"a/b=",a/b)
a=-2.50
b=1.25E2
```



```
print ("Chia của số thực và số thực")
print ("a=",a,"b=",b,"a/b=",a/b)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Chia của số nguyên và số thực
a= 10 b= -20.5 a/b= -0.4878048780487805
Chia của số thực và số thực
a= -2.50 b= 125.0 a/b= -0.02
```

## VÍ DỤ: CHIA VỚI SỐ PHỨC

Khi một trong các toán hạng là một số phức, phép chia giữa toán hạng còn lại và cả hai phần của đối tượng số phức (phần thực và ảo) đều xảy ra.



< 8 Sync Dev />

```
a=7.5+7.5j
b=2.5
print ("Chia của số phức và số thực")
print ("a =",a,"b =",b,"a/b =",a/b)
print ("a =",a,"b =",b,"b/a =",b/a)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
Chia của số phức và số thực
a = (7.5+7.5j) b = 2.5 a/b = (3+3j)
```

```
a = (7.5+7.5j) b = 2.5 b/a = (0.16666666666666666-
0.16666666666666666j)
```

Nếu tử số là 0, kết quả của phép chia luôn là 0 ngoại trừ khi mẫu số là 0, trong trường hợp này, Python ném ra ZeroDivisionError với thông báo lỗi Phép Chia Cho Không.



< 8 Sync Dev />

```
a=0
b=2.5
print ("a=",a,"b=",b,"a/b=",a/b)
print ("a=",a,"b=",b,"b/a=",b/a)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
a= 0 b= 2.5 a/b= 0.0
Traceback (most recent call last):
 File "C:\Users\mlath\examples\example.py", line 20, in
<module>
 print ("a=",a,"b=",b,"b/a=",b/a)
 ~^^
ZeroDivisionError: float division by zero
```

## TOÁN TỬ LẤY DƯ (%)

Python định nghĩa ký hiệu "%" được gọi là ký hiệu Phần Trăm, là toán tử Lấy Dư (hoặc phần dư). Nó trả về phần dư sau khi mẫu số chia tử số. Nó cũng có thể được gọi là toán tử dư dư. Kết quả của toán tử lấy dư là số còn lại sau khi chia tỷ lệ nguyên. Ví dụ, khi 10 chia 3, tỷ lệ là 3 và phần dư là 1. Do đó, 10%3 (thường được phát âm là 10 mod 3) kết quả là 1.

# VÍ DỤ: PHÉP LẤY DƯ VỚI GIÁ TRỊ SỐ NGUYÊN

Nếu cả hai toán hạng đều là số nguyên, giá trị phần dư là một số nguyên. Nếu tử số có thể chia hết, phần dư là 0. Nếu tử số nhỏ hơn mẫu số, phần dư bằng tử số. Nếu mẫu số là 0, Python ném ra `ZeroDivisionError`.



< 8 Sync Dev />

```
a=10
b=2
print ("a=",a, "b=",b, "a%b=", a%b)
a=10
b=4
print ("a=",a, "b=",b, "a%b=", a%b)
print ("a=",a, "b=",b, "b%a=", b%a)
a=0
b=10
print ("a=",a, "b=",b, "a%b=", a%b)
print ("a=", a, "b=", b, "b%a=",b%a)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
a= 10 b= 2 a%b= 0
a= 10 b= 4 a%b= 2
a= 10 b= 4 b%a= 4
a= 0 b= 10 a%b= 0
Traceback (most recent call last):
 File "C:\Users\mlath\examples\example.py", line 13, in
<module>
 print ("a=", a, "b=", b, "b%a=",b%a)
 ~^^
ZeroDivisionError: integer modulo by zero
```



# VÍ DỤ: PHÉP LẤY DƯ VỚI GIÁ TRỊ SỐ THỰC

Nếu một trong các toán hạng là một số thực, giá trị mod luôn là số thực.



< 8 Sync Dev />

```
a=10
b=2.5
print ("a=",a, "b=",b, "a%b=", a%b)
a=10
b=1.5
print ("a=",a, "b=",b, "a%b=", a%b)
a=7.7
b=2.5
print ("a=",a, "b=",b, "a%b=", a%b)
a=12.4
b=3
print ("a=",a, "b=",b, "a%b=", a%b)
```

Sẽ tạo ra đầu ra sau:



< 8 Sync Dev />

```
a= 10 b= 2.5 a%b= 0.0
a= 10 b= 1.5 a%b= 1.0
a= 7.7 b= 2.5 a%b= 0.200000000000000018
a= 12.4 b= 3 a%b= 0.400000000000000036
```