

Целочисленная арифметика

Операция деления (/) для целых чисел всегда возвращает значение типа float, даже если делимое делится нацело на делитель. Часто бывает необходимым выполнить деление с целочисленным результатом. Для этого в Python есть специальная операция целочисленного деления, которая обозначается //. Результатом будет число, которое было бы получено в результате деления, но с отброшенной дробной частью. Другая связанная с этим операция — это операция взятия остатка от деления, обозначаемая %.

```
>>> 17 / 3
5.666666666666667
>>> 17 // 3    - сколько раз 3-ка помещается в 17-ти
5
>>> 17 % 3     - целый остаток от деления 17 на 3
2
```

Еще примеры, как работает целочисленная арифметика:

7 // 3=2	7 % 3=1
13 // 4=3	13 % 4=1
17 // 13=1	17 % 13=4

В случае отрицательных чисел операции по-разному определены в различных языках программирования. Например, в Python 3 округление по-прежнему будет производиться вниз.

Например:

```
>>> (-17) // 3
-6
>>> (-17) % 3
1
```

Задача. Расписание уроков

В некоторой школе расписание составлено таким образом, что на неделе 5 учебных дней, в которые суммарно проходит n уроков. При этом количество уроков в любые два дня отличается не более чем

на 1. Определите минимальное число уроков в день и количество дней, в которых уроков больше всего.

Решение:

```
n = int(input())  
print(n // 5)  
print(n % 5)
```

Задача. Сумма цифр трехзначного числа

Дано трёхзначное число. Найдите сумму его цифр.

Решение

```
n = int(input())  
d3 = n % 10  
d2 = n // 10 % 10  
d1 = n // 100  
print(d1 + d2 + d3)
```