

Исполнитель преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1

2. Умножить на 3

Первая команда увеличивает число на экране на 1, вторая умножает его на 3. Программа для исполнителя — это последовательность команд.

Сколько существует программ, которые преобразуют исходное число 2 в число 90, и при этом траектория вычислений содержит число 28?

Траектория вычислений — это последовательность результатов выполнения всех команд программы. Например, для программы 212 при исходном числе 4 траектория будет состоять из чисел 12, 13, 39.

Решение.

Искомое количество программ равно произведению количества программ, получающих из числа 2 число 28, на количество программ, получающих из числа 28 число 90, при этом траектория вычислений должна содержать число 28.

Пусть $R(n)$ — количество программ, которые число 2 преобразуют в число n .

Верны следующие соотношения:

$R(n) = R(n-1) + R(n/3)$ (если n — кратно 3).

$R(2) = 1.$

$R(3) = 1.$

$R(4) = 1.$

$R(5) = 1.$

$R(6) = 2.$

$R(7) = 2.$

$R(8) = 2.$

$R(9) = 3.$

$R(10) = 3.$

$R(11) = 3.$

$R(12) = 4.$

$R(13) = 4.$

$R(14) = 4.$

$R(15) = 5.$

$R(16) = 5.$

$R(17) = 5.$

$R(18) = 7.$

$R(19) = 7.$

$R(20) = 7.$

$R(21) = 9.$

$R(22) = 9.$

$R(23) = 9.$

$R(24) = 11.$

$R(25) = 11.$

$R(26) = 11.$

$R(27) = 14.$

$R(28) = 14.$

Из числа 28 число 90 можно получить четырьмя способами: последовательностью команд 2111111, последовательностью команд 12111, последовательностью команд 112 и последовательностью команд 11..11.

Таким образом, количество программ, удовлетворяющих условию задачи, равно $14 \cdot 4 = 56$.

Ответ: 56.

Приведём другое решение на языке Python.

```
def f(x, y):
    if x > y:
        return 0
    if x == y:
        return 1
    else:
        return f(x + 1, y) + f(x * 3, y)
print(f(2, 28) * f(28, 90))
```