

Исполнитель РазДваТри преобразует число на экране.

У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Умножить на 2
3. Прибавить 3

Первая команда увеличивает число на экране на 1, вторая умножает его на 2, третья увеличивает на 3.

Программа для исполнителя РазДваТри — это последовательность команд.

Сколько существует программ, которые преобразуют исходное число 3 в число 16 и при этом траектория вычислений содержит число 12?

Траектория вычислений — это последовательность результатов выполнения всех команд программы. Например, для программы 312 при исходном числе 6 траектория будет состоять из чисел 9, 10, 20.

Решение.

Искомое количество программ равно произведению количества программ, получающих из числа 3 число 12, на количество программ, получающих из числа 12 число 16. Траектория вычислений должна содержать число 12.

Пусть $R(n)$ — количество программ, которые число 3 преобразуют в число n , а $F(n)$ — количество программ, которые число 12 преобразуют в число n .

Верны следующие соотношения:

$$R(n) = R(n-1) + R(n/2)(\text{если } n \text{ — чётно}) + R(n-3).$$

$$F(n) = F(n-1) + F(n/2)(\text{если } n \text{ — чётно}) + F(n-3).$$

$$R(3) = 1$$

$$R(4) = 1$$

$$R(5) = 1$$

$$R(6) = 3$$

$$R(7) = 4$$

$$R(8) = 6$$

$$R(9) = 9$$

$$R(10) = 14$$

$$R(11) = 20$$

$$R(12) = 32$$

$$F(12) = 1$$

$$F(13) = 1$$

$$F(14) = 1$$

$$F(15) = 2$$

$$F(16) = 3$$

Таким образом, количество программ, удовлетворяющих условию задачи, равно $32 \cdot 3 = 96$.

Ответ: 96.

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 * 2, y) + f(x + 3, y)
print(f(3, 12) * f(12, 16))
```