

Исполнитель Тренер преобразует число на экране.

У исполнителя есть две команды, которым присвоены номера:

### 1. Прибавить 1

### 2. Умножить на 2

Первая команда увеличивает число на экране на 1, вторая умножает его на 2. Программа для исполнителя Тренер — это последовательность команд.

Сколько существует программ, которые преобразуют исходное число 1 в число 30 и при этом траектория вычислений содержит числа 10 и 21?

Траектория должна содержать оба указанных числа. Траектория вычислений — это последовательность результатов выполнения всех команд программы. Например, для программы 212 при исходном числе 7 траектория будет состоять из чисел 14, 15, 30.

### Решение.

Нужно найти количество программ, которые из 1 получают 10, количество программ, которые из 10 получают 21, количество программ, которые из 21 получают 30 и перемножить найденные значения. Сначала найдём количество программ, получающих 10 из 1.

Обозначим  $R(n)$  — количество программ, которые преобразуют число 1 в число  $n$ .

Верны следующие соотношения:

1. Если  $n$  не делится на 2, то тогда  $R(n) = R(n - 1)$ , так как существует единственный способ получения  $n$  из  $n - 1$  — прибавление единицы.

2. Пусть  $n$  делится на 2.

Если  $n > 1$ , то  $R(n) = R(n / 2) + R(n - 1)$ .

Если  $n = 1$ , то  $R(n) = 1$  (два способа: прибавление единицы и удвоение).

Теперь можно постепенно вычислить все значения:

$$R(2) = R(1) + R(1) = 1 + 1 = 2 = R(3)$$

$$R(4) = R(2) + R(3) = 2 + 2 = 4 = R(5),$$

$$R(6) = R(3) + R(5) = 2 + 4 = 6 = R(7),$$

$$R(8) = R(4) + R(7) = 4 + 6 = 10 = R(9),$$

$$R(10) = R(5) + R(9) = 4 + 10 = 14$$

Программ, получающих из числа 10 число 21 достаточно мало, можно их просто перечислить: 21, 1111111111.

А программ, получающих из числа 21 в число 30 всего один способ: добавление единиц.

Тем самым, находим ответ:

Ответ: 28.

### 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)
print(f(1, 10) * f(10, 21) * f(21, 30))
```