

Исполнитель M17 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

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

Первая из них увеличивает число на экране на 1, вторая увеличивает его на 2, третья умножает на 3. Программа для исполнителя M17 – это последовательность команд. Сколько существует таких программ, которые преобразуют исходное число 2 в число 12 и при этом траектория вычислений программы содержит числа 8 и 10? Траектория должна содержать оба указанных числа.

Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы 132 при исходном числе 7 траектория будет состоять из чисел 8, 24, 26.

Решение.

Искомое количество программ равно произведению количества программ, получающих из числа 2 число 8, на количество программ, получающих из числа 8 число 10, и на количество программ, получающих из числа 10 число 12.

Будем решать задачу с конца. Число 12 из числа 10 можно получить двумя способами (10+1+1; 10+2). Число 10 из числа 8 можно получить двумя способами (8+1+1; 8+2). Остается узнать количество способов получения числа 8 из числа 2. Начнем свои рассуждения с числа 3, т.к. двойка это начальное число. Тройку можно получить только одним способом – прибавив 1. Четверку получим двумя способами – прибавив единицу к тройке или добавив двойку к двойке и т.д. Запишем эти рассуждения в следующем виде:

$$\begin{aligned}R(2) &= 1 \\R(3) &= R(2) = 1 \\R(4) &= R(3) + R(2) = 2 \\R(5) &= R(4) + R(3) = 2 + 1 = 3 \\R(6) &= R(5) + R(4) + R(2) = 3 + 2 + 1 = 6 \\R(7) &= R(6) + R(5) = 6 + 3 = 9 \\R(8) &= R(7) + R(6) = 9 + 6 = 15\end{aligned}$$

Таким образом, количество программ, удовлетворяющих условию задачи, равно $R(2) * R(8) * R(10) * R(12) = 1 * 15 * 2 * 2 = 60$.

Ответ: 60.

Приведём решение на языке PascalABC.

```
var a: array[-10..12] of integer;
i: integer;
begin
  for i:=2 to 8 do begin
    a[2]:=1;
    if i mod 3 = 0 then a[i]:=a[i-1]+a[i-2]+a[i div 3];
    if i mod 3 <> 0 then a[i]:=a[i-1]+a[i-2];
  end;
  for i:=2 to 7 do
    a[i]:=0;
  for i:=9 to 10 do begin
    if i mod 3 = 0 then a[i]:=a[i-1]+a[i-2]+a[i div 3];
    if i mod 3 <> 0 then a[i]:=a[i-1]+a[i-2];
  end;
  for i:=7 to 9 do
    a[i]:=0;
  for i:=11 to 12 do begin
    if i mod 3 = 0 then a[i]:=a[i-1]+a[i-2]+a[i div 3];
    if i mod 3 <> 0 then a[i]:=a[i-1]+a[i-2];
  end;
  println(a[12]);
end.
```

Приведём решение Сергея Никифорова 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(2, 8) * f(8, 10) * f(10, 12))
```