

## 23 (повышенный уровень, время – 8 мин)

Тема: динамическое программирование.

Что проверяется:

Умение анализировать результат исполнения алгоритма

*1.6.2. Вычислимость. Эквивалентность алгоритмических моделей (?).*

*1.1.3. Умение строить информационные модели объектов, систем и процессов в виде алгоритмов (?).*

Что нужно знать:

- динамическое программирование – это способ решения сложных задач путем сведения их к более простым задачам того же типа
- с помощью динамического программирования решаются задачи, которые требуют полного перебора вариантов:
  - «подсчитайте количество вариантов...»
  - «как оптимально распределить...»
  - «найдите оптимальный маршрут...»
- динамическое программирование позволяет ускорить выполнение программы за счет использования дополнительной памяти; полный перебор не требуется, поскольку запоминаются решения всех задач с меньшими значениями параметров

### Пример задания:

**Р-09 (демо- 2021).** Исполнитель 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 – это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 20, и при этом траектория вычислений содержит число 10?

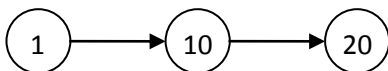
Решение (теоретическое):

- запишем рекуррентную формулу для вычисления  $K_N$  – количества возможных программ для получения числа  $N$  из некоторого начального числа:

$K_N = K_{N-1}$ , если  $N$  не делится на 2

$K_N = K_{N-1} + K_{N/2}$ , если  $N$  делится на 2

- все допустимые программы можно разбить на 2 части:
  - переход от 1 до 10
  - переход от 10 до 20



- обозначим через  $K_{a \rightarrow b}$  количество возможных программ получения числа  $b$  из числа  $a$
- очевидно, что если траектория проходит через  $c$ , то  $K_{a \rightarrow b} = K_{a \rightarrow c} \cdot K_{c \rightarrow b}$  для любого  $c$ , такого что  $a < c < b$
- поэтому  $K_{1 \rightarrow 20} = K_{1 \rightarrow 10} \cdot K_{10 \rightarrow 20}$
- вычисляем эти значения отдельно стандартным способом по рекуррентным формулам п. 1:

$N$	1	2	3	4	5	6	7	8	9	10
$K_N$	1	2	2	4	4	6	6	10	10	14

N	10	11	12	13	14	15	16	17	18	19	20
$K_N$	1	1	1	1	1	1	1	1	1	1	2

- 7) (И.В. Степанов) этот результат достаточно очевиден и без таблицы: понятно, что получить 20 из 10 с помощью заданных команд можно только двумя способами – добавляя по единице или умножив 10 на 2
- 8) перемножаем результаты двух этапов:  $14 \cdot 2 = 28$
- 9) Ответ: **28**.

#### Решение (электронные таблицы Excel, LibreOffice Calc):

- 1) главная проблема при решении этого задания – высокая вероятность арифметической ошибки, поэтому во время компьютерного экзамена можно реализовать этот же алгоритм в электронных таблицах
- 2) сначала найдём количество программ для перехода от числа 1 к числу 10; построим ряд этих чисел в первой строке и запишем начальную единицу в первую ячейку второй строки:

	A	B	C	D	E	F	G	H	I	J	K
1	N	1	2	3	4	5	6	7	8	9	10
2	KN	1									

- 3) в строке 2 будем считать значение  $K_N$  по разным формулам для чётных и нечётных N; предположим, что значение  $K_{N/2}$  записано в строке 3 того же столбца; тогда в ячейку C2 нужно записать такую формулу:

C2	fx =ЕСЛИ(ОСТАТ(С1;2)=0;В2+С3;В2)										
	A	B	C	D	E						
1	N		1	2	3	4					
2	KN		1	2							
3				1							

- 4) следующий вопрос – как получить  $K_{N/2}$ ; для этого нужно:

а) вычислить значение  $N/2$ :

**C1/2**

б) найти это значение в диапазоне B1:K1:

**ПОИСКПОЗ (C1/2 ; \$B\$1 : \$K\$1 ; 0)**

функция ПОИСКПОЗ возвращает номер элемента в массиве; в данном случае она ищет значение C1/2 в массиве B1:K1; последний аргумент 0 означает, что ищем точное совпадение

в) взять значение в строке 2 того же столбца, это и будет  $K_{N/2}$ :

**=СМЕЩ (\$A\$2 ; 0 ; ПОИСКПОЗ (C1/2 ; \$B\$1 : \$K\$1 ; 0) )**

функция СМЕЩ находит значение ячейки, которая смещена относительно ячейки A2; второй аргумент задаёт смещение по строкам (0 – в той же строке), а третий – смещение по столбцам, такое, какое вернула функция ПОИСКПОЗ

- 5) вот как выглядит формула в ячейке C3:

C3	fx =СМЕЩ(\$A\$2;0;ПОИСКПОЗ(C1/2;\$B\$1:\$K\$1;0))										
	A	B	C	D	E	F					
1	N		1	2	3	4	5				
2	KN		1	2							
3				1							

- 6) теперь копируем («протягиваем») формулы в C2 и C3 вправо до конца таблицы:

	A	B	C	D	E	F	G	H	I	J	K
1	N	1	2	3	4	5	6	7	8	9	10
2	KN	1	2	2	4	4	6	6	10	10	14
3			1	#Н/Д	2	#Н/Д	2	#Н/Д	4	#Н/Д	4

- 7) здесь #Н/Д означает, что значение  $N/2$  не найдено в первой строке (оно получилось дробное), но по формуле, которую мы записали в C2 и скопировали на всю вторую строку, для нечётных N это значение не используется

- 8) вторую часть таблицы можно построить так же, но можно просто учесть, что второе слагаемое в формуле будет только для  $N = 20$ :

L10		=СМЕЩ(\$A\$2;0;ПОИСКПОЗ(L8/2;\$B\$8:\$K\$8;0))											
	A	B	C	D	E	F	G	H	I	J	K	L	
8	N	10	11	12	13	14	15	16	17	18	19	20	
9	KN	1	1	1	1	1	1	1	1	1	1	2	
10												1	

- 9) перемножая значения последних ячеек двух таблиц, получаем  $14 \cdot 2 = 28$ .

10) Ответ: 28.

- 11) Про работу функций СМЕЩ и ПОИСКПОЗ можно посмотреть здесь:

<https://www.youtube.com/watch?v=BQN2nxLOaLg>

#### Решение (электронные таблицы OpenOffice Calc):

- 1) в электронных таблицах OpenOffice Calc нужно использовать английские названия функций:

C2: =IF (MOD (C1 ; 2) = 0 ; B2+C3 ; B2)

C3: =OFFSET (\$A\$2 ; 0 ; MATCH (C1/2 ; \$B\$1 : \$K\$1 ; 0) )

- 2) Ответ: 28.

#### Решение (рекурсивная программа, Python):

- 3) для проверки (если есть время) можно написать программу, реализующую тот же алгоритм  
 4) вычисления по рекуррентным формулам можно организовать с помощью рекурсии  
 5) рекурсивная функция, которая возвращает количество программ для преобразования числа start в число x, может быть написана так:

```
def numProg( start, x ):
    if x < start: return 0          # (1)
    if x == start: return 1         # (2)
    K = numProg( start, x-1 )       # (3)
    if x % 2 == 0:
        K += numProg( start, x//2 ) # (4)
    return K
```

- 6) если число  $x$  меньше, чем начальное значение, количество программ равно 0 (строка (1))  
 7) если число  $x$  равно начальному значению, количество программ равно 1 (строка (2))  
 8) в остальных случаях всегда учитываем количество программ предыдущего числа (если последняя команда программы будет +1), см. строку (3)  
 9) если число чётное, нужно добавить ещё и количество программ для числа  $x//2$  (строка (4))  
 10) в основной программе вычисляем количество программ от 1 до 10 и умножаем на количество программ от 10 до 20:

```
print( numProg(1,10)*numProg(10,20) )
```

- 11) Ответ: 28.

- 12) аналогичная программа на Паскале:

```
function numProg( start, x: integer ): integer;
var K: integer;
begin
    if x < start then numProg := 0
    else if x = start then numProg := 1
    else begin
        K := numProg( start, x-1 );
        if x mod 2 = 0 then
            K := K + numProg( start, x div 2 );
        numProg := K;
    end;
end;

begin
    writeln( numProg(1,10)*numProg(10,20) );
end.
```

- 13) аналогичная программа на C++:

```
#include <iostream>
```

```

using namespace std;

int numProg( int start, int x ) {
    int K;
    if( x < start ) return 0;
    if( x == start ) return 1;
    K = numProg( start, x-1 );
    if( x % 2 == 0 )
        K += numProg( start, x / 2 );
    return K;
}

int main()
{
    cout << numProg(1,10)*numProg(10,20) ;
}

```

Решение (программа на Python, А. Сидоров):

- 1) можно использовать другую идею: начав со значения **start**, мы можем попасть в **x** либо через значение **start+1**, либо через **2\*start**
- 2) получается такой краткий вариант программы:

```

def numProg( start, x ):
    if x < start: return 0
    if x == start: return 1
    return numProg(start+1,x) + numProg(start*2,x)

print( numProg(1,10)*numProg(10,20) )

```

Главное его достоинство в том, что не нужно делать проверку на чётность **x** и использовать обратные операции.

- 3) Ответ: **28**.

- 4) аналогичная программа на Паскале:

```

function numProg( start, x: integer ): integer;
var K: integer;
begin
    if x < start then numProg := 0
    else if x = start then numProg := 1
    else
        numProg := numProg(start+1,x) + numProg(start*2,x);
    end;

begin
    writeln( numProg(1,10)*numProg(10,20) );
end.

```

- 5) аналогичная программа на C++:

```

#include <iostream>
using namespace std;

int numProg( int start, int x ) {
    int K;
    if( x < start ) return 0;
    if( x == start ) return 1;
    return numProg(start+1,x) + numProg(start*2,x);
}

int main()
{
    cout << numProg(1,10)*numProg(10,20) ;
}

```

Решение (динамическое программирование, Python):

- 1) рекурсивная программа может работать очень медленно для больших чисел или при большом количестве команд; вместо этого можно применить динамическое программирование, заполняя массив по тем же формулам
- 2) создаём массив  
`TARGET = 20`  
`K = [0]*(TARGET+1)`  
 константа **TARGET** означает наибольшее число, которое нужно получить; размер массива на единицу больше, элемент `K[0]` мы использовать не будем, так удобнее (ведь нас интересуют числа, начиная с 1)
- 3) записываем в первый элемент единицу:  
`K[1] = 1`
- 4) вот процедура, которая заполняет по приведённым выше формулам элементы массива с индексами от `start+1` до `x` (элемент `K[start]` должен быть уже заполнен!)  

```
def fillArray( start, x ):
    for i in range(start+1,x+1):
        K[i] = K[i-1]
        if i % 2 == 0 and i // 2 >= start:
            K[i] += K[i//2]
```

 в условие добавилась вторая часть (`i//2 >= start`) для того, чтобы не захватывать значения `K[i]` при `i < start`
- 5) основная программа заполняет две части массива и выводит результат:  

```
fillArray( 1, 10 )
fillArray( 10, 20 )
print( K[TARGET] )
```
- 6) обратите внимание, что после первого вызова процедуры **fillArray** значение `K[10]` уже определено, так что всё готово для заполнения второй части
- 7) Ответ: **28**.
- 8) аналогичная программа на Паскале:  

```
const TARGET = 20;
var K: array[1..TARGET] of integer;

procedure fillArray( start, x: integer );
var i: integer;
begin
    for i:=start+1 to x do begin
        K[i] := K[i-1];
        if (i mod 2 = 0) and (i div 2 >= start) then
            K[i] := K[i] + K[i div 2];
        end;
    end;

begin
    K[1] := 1;
    fillArray( 1, 10 );
    fillArray( 10, 20 );
    writeln( K[TARGET] );
end.
```
- 9) аналогичная программа на C++:  

```
#include <iostream>
using namespace std;

const int TARGET = 20;
int K[TARGET+1] = {0, 1};
void fillArray( int start, int x ) {
    for( int i = start+1; i<=x; i++ ) {
        K[i] = K[i-1];
        if( i % 2 == 0 and i / 2 >= start )
            K[i] += K[i/2];
    }
}
```

```

    }
}

int main() {
    fillArray( 1, 10 );
    fillArray( 10, 20 );
    cout << K[TARGET] << endl;
}

```

#### Решение (динамическое программирование, 2 таблицы, А.Н. Носкин):

- идея решения задачи в том, чтобы получить количество решений из числа "1" в число "10" (число, через которое проходят все траектории), а потом получить количество решений из числа "10" в число "20", при этом во втором случае расчеты начинаются сначала, то есть количество решений при числе "10" равно 1;
- результатом решения задачи будет перемножения количества решений одной таблицы на количество решений второй таблицы;
- создаём массив и заполняем количеством решений для каждого числа; при этом следует учесть формулы для пересчёта количества решений:
 

```

a[i] = a[i-1] - для чисел не кратных 2;
a[i] = a[i-1] + a[i//2] - для чисел кратных 2

```
- фрагмент кода для вычисления количества решений до числа "10".
 

```

a = [0]*11 # массив от 0 - 10
# табличка от 1 до 10
a[1] = 1
for i in range(2,10+1):
    if i%2==0 and i>=2:
        a[i] = a[i-1] + a[i//2]
    else: a[i] = a[i-1]

```
- Фрагмент кода для вычисления количества решений от числа "10" до числа "20". При этом важно учесть, что команда Умножить на 2 работает с индекса 20 и более, так как  $10*2 = 20$ .
 

```

b = [0]*21 # массив от 0 - 20
# табличка от 10 до 20
b[10] = 1
for i in range(11,20+1):
    if i%2==0 and i>=20:
        b[i] = b[i-1] + b[i//2]
    else: b[i] = b[i-1]

```
- Результат решения задачи будет перемножение последних элементов массива.
 

```

print(a[10]*b[20])

```

 Вывод результата можно организовать и так:
 

```

print(a[-1]*b[-1])

```
- Ответ: **28**.

#### Ещё пример задания:

**Р-08 (демо-вариант 2018 г.).** Исполнитель М17 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Первая команда увеличивает число на экране на 1, вторая – увеличивает его на 2, а третья – умножает его на 3. Программа для исполнителя М17 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 12 и при этом траектория вычислений содержит числа 8 и 10?

#### Решение:

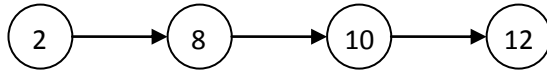
- 1) запишем рекуррентную формулу для вычисления  $K_N$  – количества возможных программ для получения числа  $N$  из некоторого начального числа:

$$K_N = K_{N-1} + K_{N-2}, \text{ если } N \text{ не делится на } 3$$

$$K_N = K_{N-1} + K_{N-2} + K_{N/3}, \text{ если } N \text{ делится на } 3$$

- 2) все допустимые программы можно разбить на 3 части:

- переход от 2 до 8
- переход от 8 до 10
- переход от 10 до 12



- 3) обозначим через  $K_{a \rightarrow b}$  количество возможных программ получения числа  $b$  из числа  $a$
- 4) очевидно, что если траектория проходит через  $c$ , то  $K_{a \rightarrow b} = K_{a \rightarrow c} \cdot K_{c \rightarrow b}$  для любого  $c$ , такого что  $a < c < b$
- 5) поэтому  $K_{2 \rightarrow 12} = K_{2 \rightarrow 8} \cdot K_{8 \rightarrow 10} \cdot K_{10 \rightarrow 12}$
- 6) вычисляем эти значения отдельно стандартным способом по рекуррентным формулам п. 1:

$N$	2	3	4	5	6	7	8
$K_N$	1	1	2	3	6	9	15

$N$	8	9	10
$K_N$	1	1	2

	10	11	12
	1	1	2

- 7) и перемножаем:  $15 \cdot 2 \cdot 2 = 60$
- 8) Ответ: **60**.

**Решение (программа на Python, Б.С. Михлин):**

- 1) полная программа:
- ```

# функция подсчета количества цепочек команд (программ) .
# x - текущее число, t - цель (target) .
def nProg( x, t ):
    if x == t: # если цель достигнута, то
        return 1 # завершить функцию, посчитав цепочку (программу)
    if x > t: # если перелет, то
        return 0 # завершить функцию, не считая цепочку
    # x < t - продолжаем строить дерево:
    return nProg( x+1, t ) + nProg( x+2, t ) + nProg( x*3, t )
print( nProg( 2, 8 ) * nProg( 8, 10 ) * nProg( 10, 12 ) )

```
- 2) Ответ: **60**.

## Ещё пример задания:

**Р-07.** Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 29 и при этом траектория вычислений содержит число 14 и не содержит числа 25?

**Решение:**

- 1) у нас в задании две особые точки – числа 14 (через которое должна проходить траектория) и 25 (а сюда она попасть НЕ должна)

- 2) сначала, так же, как и в задачах, рассмотренных ниже, составляем рекуррентную формулу, по которой будем вычислять количество  $K_N$  обозначить количество разных программ для получения числа  $N$  из начального числа:
- 3) число  $N$  могло быть получено одной из двух операций:
- увеличением на 1 числа  $N-1$ ;
  - умножением на 2 числа  $N/2$  (только для  $N$ , которые делятся на 2);

$$K_N = K_{N-1} \text{ для нечётных чисел}$$

$$K_N = K_{N-1} + K_{N/2} \text{ для чётных чисел}$$

- 4) для начального числа 2 количество программ равно 1: существует только одна пустая программа, не содержащая ни одной команды;  $K_1 = 1$ .
- 5) составляем таблицу до первой особой точки – числа 14:

| N     | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|-------|---|---|---|---|---|---|---|---|----|----|----|----|----|
| $K_N$ | 1 | 1 | 2 | 2 | 3 | 3 | 5 | 5 | 7  | 7  | 10 | 10 | 13 |

- 6) поскольку число 14 должно обязательно войти в траекторию, начинаем составлять вторую часть таблицы (до второй контрольной точки, 25) с этого числа заново, считая, что все ячейки для меньших чисел – нулевые

| N     | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| $K_N$ | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 0  |    |    |    |    |

- 7) поскольку траектория не может проходить через 25, для  $N = 25$  принимаем  $K_N = 0$  (в таблице эта ячейка выделена красным цветом)
- 8) дальше заполняем оставшиеся ячейки второй части таблицы обычным способом (см. задачи ниже):

| N     | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| $K_N$ | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 0  | 0  | 0  | 13 | 13 |

- 9) Ответ: 13.

Решение (программа на Python, Б.С. Михлин):

- 3) полная программа:
- ```
# функция подсчета количества цепочек команд (программ).
# x - текущее число, t - цель (target).
def nProg( x, t ):
    if x == t: # если цель достигнута, то
        return 1 # завершить функцию, посчитав цепочку (программу)
    if x==25 or x > t: # если x=25 или перелет, то
        return 0 # завершить функцию, не считая цепочку
    # x < t - продолжаем строить дерево:
    return nProg( x+1, t ) + nProg( x*2, t )
print( nProg( 2, 14 ) * nProg( 14, 29 ) )
```

- 4) Ответ: 13.

Решение (2 таблицы, Python, А.Н. Носкин):

- 1) идея: смотри разбор Р-09
- ```
a = [0]*15 # массив от 0 - 14
# заполняем от 2 до 14
a[2] = 1
for i in range(3,14+1):
    if i%2==0 and i>=4:
        a[i] = a[i-1] + a[i//2]
    else: a[i] = a[i-1]

b = [0]*30 # массив от 0 - 29
# заполняем от 14 до 29
b[14] = 1
for i in range(15,29+1):
    if i==25: b[25] = 0
    elif i%2==0 and i>=28:
```



```

    b[i] = b[i-1] + b[i//2]
else: b[i] = b[i-1]

```

```

print(a[14]*b[29])
# или так: print(a[-1]*b[-1])

```

2) Ответ: 13.

**Решение (динамическое программирование, Python, А.Н. Носкин):**

```

1) использование одной таблицы
a = [0]*30 # массив от 0 - 29
# заполняем от 2 до 14
a[2] = 1
for i in range(3,14+1):
    if i % 2 == 0:
        a[i] = a[i-1] + a[i//2]
    else: a[i] = a[i-1]

# заполняем с 15 до 29
for i in range(15,29+1):
    if i==25: a[25] = 0
    elif i%2 == 0 and i>=28:
        a[i] = a[i-1] + a[i//2]
    else: a[i] = a[i-1]
print( a[29] )
2) Ответ: 13.

```

### Ещё пример задания:

**Р-06.** У исполнителя Удвоитель 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 Удвоитель – это последовательность команд. Сколько существует программ, преобразующих число 4 в число 24, предпоследней командой которых является команда «1»?

**Решение:**

- 1) итак, мы знаем предпоследнюю команду – 1, при этом последняя команда может быть любая – 1 или 2
- 2) выходит, что нужно получить количество всех программ вида «\*11» и «\*12», где звёздочка обозначает любые команды
- 3) если программа заканчивается на «11», то до выполнения цепочки «11» у нас было число  $24 - 1 - 1 = 22$ ; поэтому нужно найти число программ для преобразования 4 в 22
- 4) для начального числа 1 количество программ равно 1: существует только одна пустая программа, не содержащая ни одной команды; если через  $K_N$  обозначить количество разных программ для получения числа  $N$  из начального числа 1, то  $K_1 = 1$ .
- 5) теперь рассмотрим общий случай, чтобы построить рекуррентную формулу, связывающую  $K_N$  с предыдущими элементами последовательности  $K_1, K_2, \dots, K_{N-1}$ , то есть с решениями таких же задач для меньших  $N$
- 6) число  $N$  могло быть получено одной из двух операций:
  - увеличением на 1 числа  $N-1$ ;
  - умножением на 2 числа  $N/2$  (только для  $N$ , которые делятся на 2, и таких, что  $N/2 \geq 4$ );

$$K_N = K_{N-1} \text{ для нечётных чисел}$$

$$K_N = K_{N-1} + K_{N/2} \text{ для чётных чисел, таких, что } N/2 \geq 4$$

7) составляем таблицу:

|   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |
|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|
| N | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|

|       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |    |    |    |
|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|----|----|
| $K_N$ | 1 | 1 | 1 | 1 | 2 | 2 | 3 | 3 | 4 | 4 | 5 | 5 | 7 | 7 | 9 | 9 | 12 | 12 | 15 |
|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|----|----|

- 8) теперь рассматриваем случай, когда программа заканчивается на «12», это значит, что до выполнения цепочки «12» у нас было число  $(24/2) - 1 = 11$ ; поэтому нужно найти число программ для преобразования 4 в 11, берём его из таблицы: 3
- 9) ответ к задаче – сумма двух значений, выделенных жёлтым маркером:  $15 + 3 = 18$ , поскольку мы рассмотрели все варианты программ, в которых предпоследняя команда – 1
- 10) Ответ: 18.

#### Решение (программа на Python):

- 1) полная программа:
- ```
TARGET = 4
def nProg( x ):
    if x == TARGET: # если цель достигнута, то
        return 1    # завершить функцию, посчитав программу
    if x < TARGET:  # если перелет, то
        return 0    # завершить функцию, не считая цепочку
    # x > TARGET - продолжаем строить дерево:
    n = nProg( x - 1 )
    if x % 2 == 0:
        n += nProg( x // 2 )
    return n
START = 24
print( nProg( START//2-1 ) + nProg( START-1-1 ) )
```
- 2) Ответ: 18.

#### Решение (ещё один варианты программы на Python, Б.С. Михлин):

- 1) полная программа:
- ```
def nProg( x, t ): # x - текущее число, t - цель (target)
    if x == t:
        return 1
    if x > t:
        return 0
    # x < t - строим дерево:
    return nProg( x + 1, t ) + nProg( x * 2, t )
# Т.к. предпоследняя команда +1, то 22 +1 +1 = 24, (11 +1) *2 = 24
print( nProg( 4, 22 ) + nProg( 4, 11 ) )
```
- 2) Ответ: 18.

#### Ещё пример задания:

**Р-05.** Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 21 и при этом траектория вычислений содержит число 10? Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы 121 при исходном числе 7 траектория будет состоять из чисел 8, 16, 17.

#### Решение (вариант 1):

- 1) заметим, что при выполнении любой из команд число увеличивается (не может уменьшаться)
- 2) для начального числа 1 количество программ равно 1: существует только одна пустая программа, не содержащая ни одной команды; если через  $K_N$  обозначить количество разных программ для получения числа  $N$  из начального числа 1, то  $K_1 = 1$ .

- 3) теперь рассмотрим общий случай, чтобы построить рекуррентную формулу, связывающую  $K_N$  с предыдущими элементами последовательности  $K_1, K_2, \dots, K_{N-1}$ , то есть с решениями таких же задач для меньших  $N$
- 4) число  $N$  могло быть получено одной из двух операций:
- увеличением на 1 числа  $N-1$ ;
  - умножением на 2 числа  $N/2$  (только для  $N$ , которые делятся на 2);

$$K_N = K_{N-1} \text{ для нечётных чисел}$$

$$K_N = K_{N-1} + K_{N/2} \text{ для чётных чисел}$$

- 5) поскольку траектория должна проходить через число 10, сначала выясняем, сколькими способами можно получить 10 из 1, а затем будем считать, сколько есть способов получить 21 из 10
- 6) заполняем таблицу от 1 до 10 по полученным формулам:

| N     | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8  | 9  | 10 |
|-------|---|---|---|---|---|---|---|----|----|----|
| $K_N$ | 1 | 2 | 2 | 4 | 4 | 6 | 6 | 10 | 10 | 14 |

- 7) второй этап – определяем таким же образом (и по таким же формулам!), сколько есть способов получить конечное число 21 из 10, только левую часть таблицы (от 1 до 10) мы уже не рассматриваем:

| N     | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|
| $K_N$ | 14 | 14 | 14 | 14 | 14 | 14 | 14 | 14 | 14 | 14 | 28 | 28 |

- 8) Ответ: **28**.

#### Решение (вариант 2, А.Н. Носкин):

- 1) первый этап (п. 1-6) такой же, как и в первом варианте (см. выше);
- 2) на втором этапе используем такую идею: если мы знаем количество команд, с помощью которых из начального числа 1 можно получить 10 и определим количество команд, с помощью которых из 10 можно получить конечное значение 21, останется только перемножить эти два числа – это и будет ответ
- 3) составляем таблицу для получения 21 из 10, используя те же рекуррентные формулы:

| N     | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|
| $K_N$ | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 2  | 2  |

- 4) результат –  $14 \times 2 = 28$

- 5) Ответ: **28**.

#### Решение (программа на Python, Б.С. Михлин):

- 3) полная программа:
- ```
# функция подсчета количества цепочек команд (программ) .
# x - текущее число, t - цель (target) .
def nProg( x, t ):
    if x == t: # если цель достигнута, то
        return 1 # завершить функцию, посчитав цепочку (программу)
    if x > t: # если перелет, то
        return 0 # завершить функцию, не считая цепочку
    # x < t - продолжаем строить дерево:
    return nProg( x+1, t ) + nProg( x*2, t )
print( nProg( 1, 10 ) * nProg( 10, 21 ) )
```
- 4) Ответ: **28**.

#### Ещё пример задания:

**Р-04.** Исполнитель Калькулятор преобразует число, записанное на экране. У исполнителя три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 2
3. прибавь следующее

Первая из них увеличивает число на экране на 1, вторая увеличивает это число на 2, а третья прибавляет к числу на экране число, большее на 1 (к числу 3 прибавляется 4, к числу 9 прибавляется 10 и т. д.). Программа для исполнителя Калькулятор – это последовательность команд. Сколько есть программ, которые число 2 преобразуют в число 10?

**Решение (1 способ, составление таблицы):**

- 1) заметим, что при выполнении любой из команд число увеличивается (не может уменьшаться)
- 2) для начального числа 2 количество программ равно 1: существует только одна пустая программа, не содержащая ни одной команды; если через  $K_N$  обозначить количество разных программ для получения числа  $N$  из начального числа 2, то  $K_2 = 1$ .
- 3) теперь рассмотрим общий случай, чтобы построить рекуррентную формулу, связывающую  $K_N$  с предыдущими элементами последовательности  $K_1, K_2, \dots, K_{N-1}$ , то есть с решениями таких же задач для меньших  $N$
- 4) число  $N$  могло быть получено одной из трёх операций сложения:
  - увеличением на 1 числа  $N-1$ ;
  - увеличением на 2 числа  $N-2$ ;
  - из некоторого числа  $X$  увеличением на  $X+1$  (следующее число), так что  $N = X + X + 1$ , откуда  $X = (N - 1) / 2$ ; так могут быть получены только нечетные числа;

поэтому

$$K_N = K_{N-1} + K_{N-2} \text{ для чётных чисел}$$

$$K_N = K_{N-1} + K_{N-2} + K_{(N-1)/2} \text{ для нечётных чисел}$$

- 5) остается по этой формуле заполнить таблицу для всех значений от 2 до 10:

N	2	3	4	5	6	7	8	9	10
$K_N$	1	1	2	4	6	11	17	30	47

- 6) Ответ:– 47.

**Решение (программа на Python, Б.С. Михлин):**

- 5) полная программа:
 

```
t = 10 # цель (target)
# Функция подсчета количества цепочек команд (программ) .
# x - текущее число.
def nProg( x ):
    if x == t: # если цель достигнута, то
        return 1 # завершить функцию, посчитав цепочку (программу)
    if x > t: # если перелет, то
        return 0 # завершить функцию, не считая цепочку
    # x < t - продолжаем строить дерево:
    return nProg( x + 1 ) + nProg( x + 2 ) + nProg( x * 2 + 1 )
print( nProg( 2 ) )
```

- 6) Ответ: 47.

**Ещё пример задания:**

**Р-03.** Исполнитель Май4 преобразует число, записанное на экране. У исполнителя три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 2
3. прибавь 4

Первая из них увеличивает число на экране на 1, вторая увеличивает это число на 2, а третья – на 4. Программа для исполнителя Май4 – это последовательность команд. Сколько есть программ, которые число 21 преобразуют в число 30?

**Решение (1 способ, составление таблицы):**

- 1) заметим, что при выполнении любой из команд число увеличивается (не может уменьшаться)

- 2) все числа, меньшие начального числа 21, с помощью этого исполнителя получить нельзя, для них количество программ будет равно 0
- 3) для начального числа 21 количество программ равно 1: существует только одна пустая программа, не содержащая ни одной команды; если через  $K_N$  обозначить количество разных программ для получения числа  $N$  из начального числа 21, то  $K_{21} = 1$ .
- 4) теперь рассмотрим общий случай, чтобы построить рекуррентную формулу, связывающую  $K_N$  с предыдущими элементами последовательности  $K_1, K_2, \dots, K_{N-1}$ , то есть с решениями таких же задач для меньших  $N$
- 5) любое число  $N > 21$  могло быть получено одной из трёх операций сложения соответственно из чисел  $N-1$ ,  $N-2$  и  $N-4$ , поэтому

$$K_N = K_{N-1} + K_{N-2} + K_{N-4}$$

- 6) остается по этой формуле заполнить таблицу для всех значений от 21 до 30:

N	21	22	23	24	25	26	27	28	29	30
$K_N$	1	1	2	3	6	10	18	31	55	96

- 7) ответ – 96.

**Решение (программа на Python, Б.С. Михлин):**

- 1) полная программа:
 

```
t = 30 # цель (target)
# функция подсчета количества цепочек команд (программ).
# x - текущее число.
def nProg( x ):
    if x == t: # если цель достигнута, то
        return 1 # завершить функцию, посчитав цепочку (программу)
    if x > t: # если перелет, то
        return 0 # завершить функцию, не считая цепочку
    # x < t - продолжаем строить дерево:
    return nProg( x + 1 ) + nProg( x + 2 ) + nProg( x + 4 )
print( nProg( 21 ) )
```
- 2) Ответ: 96.

### Ещё пример задания:

**Р-02.** У исполнителя Утроитель две команды, которым присвоены номера:

1. прибавь 1
2. умножь на 3

Первая из них увеличивает число на экране на 1, вторая – утраивает его.

Программа для Утроителя – это последовательность команд.

Сколько есть программ, которые число 1 преобразуют в число 20?

**Решение (1 способ, составление таблицы):**

- 1) заметим, что при выполнении любой из команд число увеличивается (не может уменьшаться)
- 2) начнем с простых случаев, с которых будем начинать вычисления: для чисел 1 и 2, меньших, чем 3, существует только одна программа, состоящая только из команд сложения; если через  $K_N$  обозначить количество разных программ для получения числа  $N$  из 1, то  $K_1 = K_2 = 1$ .
- 3) теперь рассмотрим общий случай, чтобы построить рекуррентную формулу, связывающую  $K_N$  с предыдущими элементами последовательности  $K_1, K_2, \dots, K_{N-1}$ , то есть с решениями таких же задач для меньших  $N$
- 4) если число  $N$  не делится на 3, то оно могло быть получено только последней операцией сложения, поэтому  $K_N = K_{N-1}$
- 5) если  $N$  делится на 3, то последней командой может быть как сложение, так и умножение
- 6) поэтому для получения  $K_N$  нужно сложить  $K_{N-1}$  (количество программ с последней командой сложения) и  $K_{N/3}$  (количество программ с последней командой умножения). В итоге получаем:

если N не делится на 3:  $K_N = K_{N-1}$

если N делится на 3:  $K_N = K_{N-1} + K_{N/3}$

7) остается заполнить таблицу для всех значений от 1 до N:

N	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$K_N$	1	1	2	2	2	3	3	3	5	5	5	7	7	7	9	9	9	12	12	12

8) Заметим, что количество вариантов меняется только в тех столбцах, где N делится на 3, поэтому из всей таблицы можно оставить только эти столбцы:

N	1	3	6	9	12	15	18	21
$K_N$	1	2	3	5	7	9	12	15

9) заданное число 20 попадает в последний интервал (от 18 до 21), поэтому ...

10) ответ – **12**.

**Решение (2 способ, подстановка – вычисления по формулам «с конца»):**

1) п. 1-6 выполняются так же, как и при первом способе; главная задача – получить рекуррентную формулу:

если N не делится на 3:  $K_N = K_{N-1}$

если N делится на 3:  $K_N = K_{N-1} + K_{N/3}$

с начальными условиями  $K_1 = K_2 = 1$

2) начинаем с заданного конечного числа 20; применяем первую формулу ( $K_N = K_{N-1}$ ), пока не дойдем до числа, делящегося на 3 (это 18):

$$K_{20} = K_{19} = K_{18}$$

3) далее применяем вторую формулу ( $K_N = K_{N-1} + K_{N/3}$ ):

$$K_{20} = K_{18} = K_{17} + K_6$$

4) применяем первую формулу для 17:

$$K_{17} = K_{16} = K_{15} \Rightarrow K_{20} = K_{15} + K_6$$

5) применяем вторую формулу для обоих слагаемых:

$$K_{20} = (K_{14} + K_5) + (K_5 + K_2) = K_{14} + 2K_5 + 1$$

где учтено, что  $K_2 = 1$

6) с помощью первой формулы переходим в правой части к числам, делящимся на 3:

$$K_{20} = K_{12} + 2K_3 + 1$$

а затем применяем вторую формулу для каждого слагаемого

$$K_{20} = (K_{11} + K_4) + 2(K_2 + K_1) + 1 = K_{11} + K_4 + 2(1+1) + 1 = K_{11} + K_4 + 5$$

7) снова используем первую формулу

$$K_{20} = K_9 + K_3 + 5$$

а затем – вторую:

$$K_{20} = (K_8 + K_3) + (K_2 + K_1) + 5 = K_8 + 2(K_2 + K_1) + 5 = K_8 + 9$$

8) и еще раз

$$K_{20} = K_6 + 9 = K_5 + K_2 + 9 = K_5 + 10 = K_3 + 10 = 2 + 10 = 12$$

9) ответ – **12**.

**Решение (3 способ, О.В. Щецова, лицей № 6, г. Дубна):**

1) будем составлять таблицу из трех столбцов: в первом записывается получаемое число от 1 до 20, во втором – какой последней командой может быть получено это число, а в третьем вычисляем количество различных программ для получения этого числа из 1

2) очевидно, что число 1 может быть получено с помощью одной единственной (пустой) программы:

Число	Как можно получить?	Количество программ
1		1

3) число 2 не делится на 3, поэтому его можно получить только командой сложения (+1), значит, количество программ для 2 совпадает с количеством программ для 1:

Число	Как можно получить?	Количество программ
1		1
2	+1	= 1

- 4) число 3 делится на 3, поэтому его можно получить с помощью двух команд: +1 (из 2) и \*3 (из 1):

Число	Как можно получить?	Количество программ
1		1
2	+1	1
3	+1 *3	1 + 1 = 2

- 5) числа 4 и 5 не делятся на 3, поэтому их можно получить только с помощью команды +1, а число 6 может быть получено двумя командами:

Число	Как можно получить?	Количество программ
1		1
2	+1	1
3	+1 *3	1 + 1 = 2
4	+1	2
5	+1	2
6	+1 *3	2 + 1 = 3

6) следующая группа – 7, 8 (не делятся на 3) и 9 (делится на 3):

Число	Как можно получить?	Количество программ
1		1
2	+1	1
3	+1 *3	1 + 1 = 2
4	+1	2
5	+1	2
6	+1 *3	2 + 1 = 3
7	+1	3
8	+1	3
9	+1 *3	3 + 2 = 5

7) далее – 10, 11 и 12:

Число	Как можно получить?	Количество программ
1		1
2	+1	1
3	+1 *3	1 + 1 = 2
4	+1	2
5	+1	2
6	+1 *3	2 + 1 = 3
7	+1	3
8	+1	3
9	+1 *3	3 + 2 = 5
10	+1	5
11	+1	5
12	+1 *3	5 + 2 = 7

8) и так далее, вот полностью заполненная таблица (до конечного числа 20):

Число	Как можно получить?	Количество программ
1		1
2	+1	1
3	+1 *3	1 + 1 = 2
4	+1	2
5	+1	2
6	+1 *3	2 + 1 = 3
7	+1	3
8	+1	3
9	+1 *3	3 + 2 = 5
10	+1	5
11	+1	5
12	+1 *3	5 + 2 = 7
13	+1	7
14	+1	7
15	+1 *3	7 + 2 = 9
16	+1	9
17	+1	9
18	+1 *3	9 + 3 = 12
19	+1	12
20	+1	12

9) ответ – количество программ, с помощью которых можно получить число 20 из 1, – считываем из последней ячейки третьего столбца

10) ответ – **12**.

**Решение (4 способ, М.В. Кузнецова и её ученики, г. Новокузнецк):**

- 1) пусть  $N$  – искомое конечное число,  $K(N)$  количества программ получения числа  $N$
- 2) тогда для построения рекуррентной формулы определения  $K(N)$ , нужно знать 2 факта:
  - а) какой может быть последняя команда и сколько есть видов этого последнего действия?



б) для каждого «последнего» действия нужно знать число программ получения предыдущего числа, сумма этих количеств и есть искомое значение  $K(N)$  – число программ получения числа  $N$ .

Например, общее количество программ получения числа 6 с помощью Утроителя равно  $K(6) = K(5) + K(2)$ , т.к. есть два способа завершения программ получения этого значения:  $6=5+1$  и  $6=2 \cdot 3$ .

- 3) число программ получения числа  $N$  зависит от числа программ получения предыдущего значения, и что программы получения чисел, кратных 3-м могут завершаться 2-мя способами:  $(N-1)+1$  или  $(N/3) \cdot 3$ , а все остальные числа получают только первым способом:  $(N-1)+1$ .
- 4) составим рекуррентную формулу для определения числа программ получения числа  $N$ :  
 при  $N=1$  имеем  $K(1) = 1$   
 если  $N$  не кратно 3:  $K(N) = K(N-1)$   
 если  $N$  делится на 3:  $K(N) = K(N-1) + K(N/3)$
- 5) с помощью этих формул заполняем таблицу следующим образом:
- в первом столбце записываем все натуральные числа от 1 до заданного  $N$ ;
  - во втором столбце – числа, на единицу меньшие (из которых может быть получено  $N$  последней операцией сложения с 1);
  - в третьем столбце для чисел, кратных 3-м, записываем частное от деления числа, записанного в первом столбце, на 3 (из этого числа может быть получено  $N$  последней операцией умножения на 3);
  - в последнем столбце вычисляем  $K(N)$ , складывая соответствующие значения для тех строк, номера которых записаны во втором и третьем столбцах:

N	N-1	N/3	K(N)
1	←		1
2	← 1		1
3	← 2	1	1+1=2
4	3		2
5	4		2
6	← 5	2	2+1=3
7	6		3
8	7		3
9	8	3	3+2=5
10	9		5
11	10		5
12	11	4	5+2=7
13	12		7
14	13		7
15	14	5	7+2=9
16	15		9
17	16		9
18	17	6	9+3=12
19	18		12
20	19		12

6) ответ – 12.

Решение (5 способ, А. Сидоров):

- 1) основная идея – число программ, преобразующих начальное число 1 в конечное 20 с помощью заданных в условии команд, равно числу программ, преобразующих конечное число 20 в начальное 1 с помощью обратных команд: «**вычти 1**» и «**раздели на 3**»
- 2) будем строить «обратное дерево» – дерево всех способов преобразования **конечного числа в начальное**; это лучше (в сравнении с построением «прямого» дерева, от начального числа к конечному), потому что операция умножения необратима – каждое число можно умножить на 3, но не каждое можно разделить на 3; из-за этого сразу отбрасываются тупиковые ветви, не дающие новых решений
- 3) рисуем сокращенное дерево, в котором черные стрелки показывают действие первой команды («прибавь 1»), а красные – действие второй команды («умножь на 3»); красные стрелки подходят только к тем числам, которые делятся на 3:

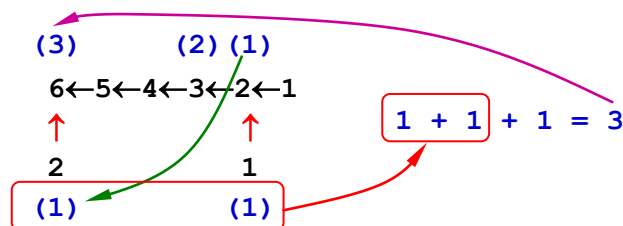
20←19←18←17←16←15←14←13←12←11←10←9←8←7←6←5←4←3←2←1

↑      ↑      ↑      ↑      ↑      ↑  
 6      5      4      3      2      1

- 4) чтобы получить количество программ для каждого числа из верхней строки, нужно сложить соответствующие количества программ для всех чисел из нижнего ряда, которые не больше данного (программы с умножением), и добавить 1 (программа, состоящая из одних сложений)
- 5) очевидно, что для получения 1 существует одна (пустая) программа; тогда для числа 2 тоже получается одна программа, а для числа 3 – две программы:

(2) (1)  
 3←2←1  
↑  
 1  
 (1)

- 6) далее, для чисел 4 и 5 получаем 2 программы (после числа 3 нет «разветвлений» – подходящих красных стрелок), а для числа 6 – 3 программы, так как «подошло» еще одно разветвление (6 можно получить умножением 2 на 3), а для числа 2 мы уже подсчитали количество программ – оно равно 1:



- 7) находить число программ для следующих чисел нам уже не понадобится, потому что при умножении на 3 они дают числа, большие, чем заданное конечное число 20
- 8) запишем полученные результаты в самой нижней строке для всех множителей от 1 до 6:

(3)      (2) (1)  
 20←19←18←17←16←15←14←13←12←11←10←9←8←7←6←5←4←3←2←1  
↑      ↑      ↑      ↑      ↑      ↑  
 6      5      4      3      2      1  
(3)      (2)      (2)      (2)      (1)      (1)

- 9) теперь остается сложить все числа в скобках в нижнем ряду (количество программ с командами умножения) и добавить 1 (одна программа, состоящая только из команд сложения):

$$3 + 2 + 2 + 2 + 1 + 1 + 1 = 12$$

- 10) ответ – **12**.

**Возможные проблемы:**

- неверно определенные начальные условия
- неверно выведенная рекуррентная формула
- ошибки при заполнении таблицы (невнимательность)
- второй способ (подстановка), как правило, приводит к большому количеству вычислений; конечно, можно отдельно выписывать все полученные ранее значения  $K_N$ , но тогда мы фактически придем к табличному методу

**Решение (программа на Python, Б.С. Михлин):**

- 1) полная программа:  

```

t = 20 # цель (target)
# функция подсчета количества цепочек команд (программ) .
# x - текущее число.
def nProg( x ):
    if x == t: # если цель достигнута, то
        return 1 # завершить функцию, посчитав цепочку (программу)
    if x > t: # если перелет, то
        return 0 # завершить функцию, не считая цепочку
    # x < t - продолжаем строить дерево:
    return nProg( x + 1 ) + nProg( x * 3 )
print( nProg( 1 ) )

```
- 2) Ответ: 12.

**Еще пример задания:**

**Р-01.** У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. увеличь вторую с конца цифру на 1

Первая из них увеличивает число на экране на 1, вторая – увеличивает на 1 число десятков.

Если перед выполнением команды 2 вторая с конца цифра равна 9, она не изменяется.

Программа для Калькулятора – это последовательность команд.

Сколько есть программ, которые число 15 преобразуют в число 28?

**Решение (1 способ, составление таблицы):**

- 1) заметим, что при выполнении любой из команд число увеличивается (не может уменьшаться)
- 2) при заданных командах очередное число  $N$  может быть получено двумя способами:
- 3) увеличением на 1 (для всех чисел, больших начального числа)
- 4) увеличением числа десятков на 1 (то есть, фактически командой «+10») – для всех чисел, больших или равных 25; например, число 24 не может быть получено этой командой ( $14 + 10 = 24$ ), потому что число 14 меньше, чем начальное значение 15
- 5) таким образом, рекуррентные формулы принимают вид  

$$K_N = K_{N-1} \text{ для всех чисел, меньших, чем 25}$$

$$K_N = K_{N-1} + K_{N-10} \text{ для чисел, больших или равных 25}$$
- 6) других способов получения числа с помощью исполнителя с заданными командами нет, то есть мы таким образом рассматриваем все возможные программы
- 7) начальное значение:  $K_{15} = 1$  (число 15 можно получить единственной пустой программой)
- 8) далее заполняем таблицу:

$N$	15	16	17	18	19	20	21	22	23	24	25	26	27	28
$K_N$	1	1	1	1	1	1	1	1	1	1	2	3	4	5

- 9) Ответ: 5

**Решение (программа на Python, Б.С. Михлин):**

- 1) полная программа:
- ```

t = 28 # цель (target)
# Функция подсчета количества цепочек команд (программ).
# x - текущее число.
def nProg( x ):
    if x == t: # если цель достигнута, то
        return 1 # завершить функцию, посчитав цепочку (программу)
    if x > t: # если перелет, то
        return 0 # завершить функцию, не считая цепочку
    # x < t - продолжаем строить дерево:
    x1 = x
    if x1 % 100 // 10 < 9: # число десятков < 9?
        x1 += 10
    return nProg( x + 1 ) + nProg( x1 )
print( nProg( 15 ) )

```
- 2) Ответ: **5**.

### Еще пример задания:

**P-00.** У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. увеличь две младшие цифры на 1

Первая из них увеличивает число на экране на 1, вторая – увеличивает на 1 число десятков и число единиц. Если перед выполнением команды 2 какая-либо из двух младших цифр равна 9, она не изменяется. Программа для Калькулятора – это последовательность команд.

Сколько есть программ, которые число 23 преобразуют в число 48?

#### Решение (1 способ, составление таблицы):

- 1) заметим, что при выполнении любой из команд число увеличивается (не может уменьшаться)
- 2) при заданных командах очередное число N может быть получено двумя способами:
- 3) увеличением на 1 (для всех чисел, больших начального числа)
- 4) увеличением обеих цифр на 1 в результате выполнения команды 2 (то есть, фактически командой «+11») – для всех чисел, больших или равных 23 + 11 = 34, которые **НЕ** оканчиваются на 0;
- 5) увеличением *только младшей* цифры на 1 в результате выполнения команды 2 (то есть, фактически командой «+1») – для всех чисел от 91 до 99, но в нашем диапазоне (23..48) таких нет
- 6) увеличением *только старшей* цифры на 1 в результате выполнения команды 2 (то есть, фактически командой «+10») – для всех чисел, больших 34 и имеющих 9 на конце; в нашем случае под этот вариант подходит только число 39
- 7) таким образом, рекуррентные формулы принимают вид

$$K_N = K_{N-1} \text{ для всех чисел, меньших, чем 34, а также для всех чисел, оканчивающихся на 0}$$

$$K_N = K_{N-1} + K_{N-11} \text{ для чисел, больших или равных 34, кроме 39}$$

$$K_N = K_{N-1} + K_{N-11} + K_{N-10} \text{ для числа 39}$$

- 8) других способов получения числа с помощью исполнителя с заданными командами нет, то есть мы таким образом рассматриваем все возможные программы
- 9) начальное значение:  $K_{23} = 1$  (число 23 можно получить единственной пустой программой)
- 10) далее заполняем таблицу:

|       |    |     |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|-------|----|-----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| N     | 23 | ... | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 |
| $K_N$ | 1  | ... | 1  | 2  | 3  | 4  | 5  | 6  | 8  | 8  | 9  | 10 | 11 | 12 | 14 | 17 | 21 | 26 |

здесь многоточия означают, что для всех чисел от 23 до 33 включительно количество программ равно 1;

- 11) например, для числа 47 количество программ вычисляется как

$$K_{47} = K_{46} + K_{36} = 17 + 4 = 21$$

а для числа 39 –как

$$K_{39} = K_{38} + K_{28} + K_{29} = 6 + 1 + 1 = 8$$

12) Ответ: 26

Решение (программа на Python, Б.С. Михлин):

3) полная программа:

```
t = 48 # цель (target)
# Функция подсчета количества цепочек команд (программ).
# x - текущее число.
def nProg( x ):
    if x == t: # если цель достигнута, то
        return 1 # завершить функцию, посчитав цепочку (программу)
    if x > t: # если перелет, то
        return 0 # завершить функцию, не считая цепочку
    # x < t - продолжаем строить дерево:
    x1 = x
    if x1 % 100 // 10 < 9: # число десятков < 9?
        x1 += 10
    if x1 % 10 < 9: # число единиц < 9?
        x1 += 1
    return nProg( x + 1 ) + nProg( x1 )
print( nProg( 23 ) )
```

4) Ответ: 26.

## Задачи для тренировки<sup>1</sup>:

- 1) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. умножь на 2

Сколько есть программ, которые число 1 преобразуют в число 16?

- 2) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. умножь на 4

Сколько есть программ, которые число 1 преобразуют в число 55?

- 3) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. умножь на 2
3. умножь на 3

Сколько есть программ, которые число 1 преобразуют в число 18?

- 4) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. умножь на 2
3. умножь на 4

Сколько есть программ, которые число 1 преобразуют в число 17?

- 5) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. умножь на 3
3. умножь на 4

Сколько есть программ, которые число 1 преобразуют в число 25?

- 6) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 2
3. умножь на 3

Сколько есть программ, которые число 1 преобразуют в число 12?

- 7) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 3
3. умножь на 2

Сколько есть программ, которые число 1 преобразуют в число 15?

- 8) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 3
3. умножь на 3

Сколько есть программ, которые число 1 преобразуют в число 15?

- 9) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 3
3. умножь на 4

Сколько есть программ, которые число 1 преобразуют в число 18?

- 10) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1

<sup>1</sup> Источники заданий:

1. Демонстрационные варианты ЕГЭ 2012-2016 гг.
2. Тренировочные работы МИОО.

- 2. прибавь 2
- 3. умножь на 4

Сколько есть программ, которые число 1 преобразуют в число 13?

- 11) У исполнителя Калькулятор две команды, которым присвоены номера:

- 1. прибавь 1
- 2. умножь на 4

Сколько есть программ, которые число 1 преобразуют в число 32?

- 12) (С.Э. Назаренко) У исполнителя Калькулятор две команды, которым присвоены номера:

- 1. прибавь 2
- 2. умножь на 2

Сколько есть программ, которые число 1 преобразуют в число 24?

- 13) (С.Э. Назаренко) У исполнителя Калькулятор две команды, которым присвоены номера:

- 1. прибавь 1
- 2. умножь на 3

Сколько есть программ, которые число 5 преобразуют в число 49?

- 14) (С.Э. Назаренко) У исполнителя Калькулятор две команды, которым присвоены номера:

- 1. прибавь 3
- 2. умножь на 3

Сколько есть программ, которые число 5 преобразуют в число 27?

- 15) (С.Э. Назаренко) У исполнителя Калькулятор три команды, которым присвоены номера:

- 1. прибавь 1
- 2. прибавь 3
- 3. умножь на 2

Сколько есть программ, которые число 3 преобразуют в число 15?

- 16) (Т.В. Белова) У исполнителя Калькулятор три команды, которым присвоены номера:

- 1. прибавь 1
- 2. умножь на 2
- 3. возведи в квадрат

Сколько есть программ, которые число 2 преобразуют в число 38?

- 17) (Т.В. Белова) У исполнителя Калькулятор три команды, которым присвоены номера:

- 1. прибавь 1
- 2. прибавь 3
- 3. возведи в квадрат

Сколько есть программ, которые число 2 преобразуют в число 19?

- 18) (Т.В. Белова) У исполнителя Калькулятор три команды, которым присвоены номера:

- 1. прибавь 1
- 2. умножь на 2
- 3. возведи в квадрат

Сколько есть программ, которые число 2 преобразуют в число 27?

- 19) У исполнителя Калькулятор две команды, которым присвоены номера:

- 1. прибавь 1
- 2. увеличь число десятков на 1

Например: при помощи команды 2 число 23 преобразуется в 33. Если перед выполнением команды 2 вторая с конца цифра равна 9, она не изменяется.

Сколько есть программ, которые число 11 преобразуют в число 27?

- 20) У исполнителя Калькулятор две команды, которым присвоены номера:

- 1. прибавь 1
- 2. увеличь число десятков на 1

Например: при помощи команды 2 число 23 преобразуется в 33. Если перед выполнением команды 2 вторая с конца цифра равна 9, она не изменяется.

Сколько есть программ, которые число 12 преобразуют в число 36?

21) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. увеличь каждый разряд числа на 1

Например, число 23 с помощью команды 2 превратится в 34, а 29 в 39 (так как младший разряд нельзя увеличить). Если перед выполнением команды 2 какая-либо цифра равна 9, она не изменяется. Сколько есть программ, которые число 25 преобразуют в число 51?

22) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. увеличь каждый разряд числа на 1

Например, число 23 с помощью команды 2 превратится в 34, а 29 в 39 (так как младший разряд нельзя увеличить). Если перед выполнением команды 2 какая-либо цифра равна 9, она не изменяется. Сколько есть программ, которые число 24 преобразуют в число 46?

23) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. увеличь каждый разряд числа на 1

Например, число 23 с помощью команды 2 превратится в 34 а 29 в 39 (так как младший разряд нельзя увеличить). Программа для Калькулятора – это последовательность команд. Сколько существует программ, которые число 26 преобразуют в число 49?

24) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. увеличь число десятков на 1

Например: при помощи команды 2 число 23 преобразуется в 33. Если перед выполнением команды 2 вторая с конца цифра равна 9, она не изменяется.

Сколько есть программ, которые число 10 преобразуют в число 33?

25) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 2
2. умножь на 2

Сколько есть программ, которые число 2 преобразуют в число 40?

26) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 3
2. умножь на 2

Сколько есть программ, которые число 3 преобразуют в число 42?

27) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. прибавь 3

Сколько есть программ, которые число 1 преобразуют в число 15?

28) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. прибавь 3

Сколько есть программ, которые число 7 преобразуют в число 20?

29) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. умножь на 2
3. умножь на 3

Сколько есть программ, которые число 1 преобразуют в число 14?

30) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 2
2. умножь на 3

Сколько есть программ, которые число 1 преобразуют в число 49?



31) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 2
2. умножь на 3

Сколько есть программ, которые число 1 преобразуют в число 55?

32) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. умножь на 1,5

Первая из них увеличивает на 1 число на экране, вторая увеличивает это число в 1,5 раза, если число чётное. К нечётным числам вторая команда неприменима. Сколько есть программ, которые число 1 преобразуют в число 20?

33) У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1
2. умножь на 1,5

Первая из них увеличивает на 1 число на экране, вторая увеличивает это число в 1,5 раза, если число чётное. К нечётным числам вторая команда неприменима. Сколько есть программ, которые число 2 преобразуют в число 22?

34) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. сделай чётное
3. сделай нечётное

Первая из них увеличивает на 1 число на экране, вторая умножает это число на 2, третья переводит число  $x$  в число  $2x + 1$ . Например, вторая команда переводит число 10 в число 20, а третья переводит число 10 в число 21. Программа для исполнителя – это последовательность команд. Сколько существует программ, которые число 2 преобразуют в число 16?

35) У исполнителя Калькулятор четыре команды, которым присвоены номера:

1. прибавь 1
2. сделай чётное
3. сделай нечётное
4. умножь на 10

Первая из них увеличивает на 1 число на экране, вторая умножает это число на 2, третья переводит число  $x$  в число  $2x + 1$ , четвертая умножает на 10. Например, вторая команда переводит число 10 в число 20, а третья переводит число 10 в число 21. Программа для исполнителя – это последовательность команд. Сколько существует программ, которые число 1 преобразуют в число 15?

36) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 2
3. прибавь 5

Программа для исполнителя – это последовательность команд. Сколько существует программ, которые число 21 преобразуют в число 30?

37) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 3
3. прибавь 6

Программа для исполнителя – это последовательность команд. Сколько существует программ, которые число 21 преобразуют в число 30?

38) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 2
2. прибавь 3
3. прибавь 5

Программа для исполнителя – это последовательность команд. Сколько существует программ, которые число 20 преобразуют в число 35?

39) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 4
3. прибавь 5

Программа для исполнителя – это последовательность команд. Сколько существует программ, которые число 30 преобразуют в число 46?

40) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 2
2. прибавь 4
3. прибавь 5

Программа для исполнителя – это последовательность команд. Сколько существует программ, которые число 31 преобразуют в число 51?

41) У исполнителя Калькулятор три команды, которым присвоены номера:

1. прибавь 1
2. прибавь 2
3. прибавь предыдущее

Первая команда увеличивает число на экране на 1, вторая увеличивает это число на 2, третья прибавляет к числу на экране число, меньшее на 1 (к числу 3 прибавляется 2, к числу 11 прибавляется 10 и т. д.). Программа для исполнителя – это последовательность команд. Сколько существует программ, которые число 2 преобразуют в число 9?

42) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2

Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 5 результатом является число 15 и при этом траектория вычислений содержит число 10?

43) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 3

Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 3 результатом является число 20 и при этом траектория вычислений содержит число 12?

44) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2
3. Прибавить 3

Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 4 результатом является число 15 и при этом траектория вычислений содержит число 8?

45) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 4 результатом является число 20 и при этом траектория вычислений содержит число 10?

- 46) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 28 и при этом траектория вычислений содержит число 7?

- 47) Исполнитель A13S преобразует целое число, записанное на экране. У исполнителя три команды, каждой команде присвоен номер:

1. Прибавь 1
2. Прибавь 3
3. Прибавь предыдущее

Первая команда увеличивает число на экране на 1, вторая увеличивает это число на 3, третья прибавляет к числу на экране число, меньшее на 1 (к числу 3 прибавляется 2, к числу 11 прибавляется 10 и т. д.). Программа для исполнителя A13S – это последовательность команд. Сколько существует программ, которые число 2 преобразуют в число 10?

- 48) Исполнитель A12S преобразует целое число, записанное на экране. У исполнителя три команды, каждой команде присвоен номер:

1. Прибавь 1
2. Прибавь 2
3. Прибавь предыдущее

Первая команда увеличивает число на экране на 1, вторая увеличивает это число на 2, третья прибавляет к числу на экране число, меньшее на 1 (к числу 3 прибавляется 2, к числу 11 прибавляется 10 и т. д.). Программа для исполнителя A12S – это последовательность команд. Сколько существует программ, которые число 3 преобразуют в число 10?

- 49) Исполнитель A23S преобразует целое число, записанное на экране. У исполнителя три команды, каждой команде присвоен номер:

1. Прибавь 2
2. Прибавь 3
3. Прибавь предыдущее

Первая команда увеличивает число на экране на 2, вторая увеличивает это число на 3, третья прибавляет к числу на экране число, меньшее на 1 (к числу 3 прибавляется 2, к числу 11 прибавляется 10 и т. д.). Программа для исполнителя A23S – это последовательность команд. Сколько существует программ, которые число 2 преобразуют в число 11?

- 50) Исполнитель Калькулятор преобразует целое число, записанное на экране. У исполнителя две команды, каждой команде присвоен номер:

1. Прибавь 1
2. Умножь на 2

Первая команда увеличивает число на экране на 1, вторая увеличивает это число в 2 раза. Сколько существует программ, которые число 3 преобразуют в число 20 и в которых предпоследняя команда 1?

- 51) Исполнитель Калькулятор преобразует целое число, записанное на экране. У исполнителя две команды, каждой команде присвоен номер:

1. Прибавь 1
2. Умножь на 2

Первая команда увеличивает число на экране на 1, вторая увеличивает это число в 2 раза. Сколько существует программ, которые число 5 преобразуют в число 32 и в которых предпоследняя команда 1?

- 52) Исполнитель Калькулятор преобразует целое число, записанное на экране. У исполнителя две команды, каждой команде присвоен номер:

1. Прибавь 1
2. Прибавь 2

Первая команда увеличивает число на экране на 1, вторая увеличивает – на 2. Сколько существует программ, которые число 4 преобразуют в число 14 и в которых предпоследняя команда 1?

- 53) Исполнитель Калькулятор преобразует целое число, записанное на экране. У исполнителя две команды, каждой команде присвоен номер:

1. Прибавь 1
2. Прибавь 2

Первая команда увеличивает число на экране на 1, вторая увеличивает – на 2. Сколько существует программ, которые число 3 преобразуют в число 18 и в которых предпоследняя команда 2?

- 54) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 21 и при этом траектория вычислений содержит число 10?

- 55) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 34 и при этом траектория вычислений содержит число 12?

- 56) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 34 и при этом траектория вычислений содержит число 10 и не содержит число 28?

- 57) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 3 результатом является число 30 и при этом траектория вычислений содержит число 20 и не содержит число 12?

- 58) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 28 и при этом траектория вычислений содержит число 25 и не содержит число 10?

- 59) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 4 результатом является число 50 и при этом траектория вычислений содержит число 6 и не содержит число 12?

- 60) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 40 и при этом траектория вычислений содержит число 20 и не содержит число 8?

- 61) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 2
2. Умножить на 3

Первая команда увеличивает число на экране на 2, вторая умножает его на 3. Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 63 и при этом траектория вычислений содержит число 25 и не содержит число 6?

- 62) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 3 результатом является число 55 и при этом траектория вычислений содержит число 18 и не содержит число 12?

- 63) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 5 результатом является число 60 и при этом траектория вычислений содержит число 8 и не содержит число 22?

- 64) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 3 результатом является число 45 и при этом траектория вычислений содержит число 10 и не содержит число 15?

- 65) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 5

Первая команда увеличивает число на экране на 1, вторая увеличивает его на 5. Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 26 и при этом траектория вычислений содержит число 15 и не содержит число 10?

- 66) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 3

Первая команда увеличивает число на экране на 1, вторая увеличивает его на 3. Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 3 результатом является число 21 и при этом траектория вычислений содержит число 12 и не содержит число 18?

- 67) Исполнитель Июнь15 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 3

Первая команда увеличивает число на экране на 1, вторая увеличивает его на 3. Программа для исполнителя Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 5 результатом является число 25 и при этом траектория вычислений содержит число 15 и не содержит число 12?

- 68) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 31 и при этом траектория вычислений содержит число 15 и не содержит число 22?

- 69) Исполнитель Июнь15 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 Июнь15 – это последовательность команд. Сколько существует программ, для которых при исходном числе 2 результатом является число 33 и при этом траектория вычислений содержит число 16 и не содержит число 30?

- 70) Исполнитель Май16 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2

Первая команда увеличивает число на экране на 1, вторая увеличивает его на 2. Сколько существует программ, для которых при исходном числе 1 результатом является число 12 и при этом траектория вычислений содержит число 7?

- 71) Исполнитель Май16 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2

Первая команда увеличивает число на экране на 1, вторая увеличивает его на 2. Сколько существует программ, для которых при исходном числе 1 результатом является число 13 и при этом траектория вычислений содержит число 7?

- 72) Исполнитель Июнь16 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 3 результатом является число 13 и при этом траектория вычислений содержит число 10?

- 73) Исполнитель Июнь16 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 2 результатом является число 12 и при этом траектория вычислений содержит число 10?

- 74) Исполнитель Июнь16 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 2 результатом является число 16 и при этом траектория вычислений содержит число 14?

- 75) Исполнитель Июнь16 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 3 результатом является число 13 и при этом траектория вычислений не содержит число 8?

- 76) Исполнитель Июнь16 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 2 результатом является число 12 и при этом траектория вычислений не содержит число 10?

- 77) Исполнитель Июнь16 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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



Сколько существует программ, для которых при исходном числе 2 результатом является число 16 и при этом траектория вычислений не содержит число 14?

- 78) Исполнитель Июнь16 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 2 результатом является число 13 и при этом траектория вычислений содержит число 10?

- 79) Исполнитель Июнь16 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 2 результатом является число 25 и при этом траектория вычислений не содержит число 20?

- 80) Исполнитель Июнь17 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 2 результатом является число 31 и при этом траектория вычислений не содержит число 25?

- 81) Исполнитель Июнь17 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Сделай нечётное

Выполняя первую команду, исполнитель увеличивает число на 1, а выполняя вторую – из числа  $x$  получает число  $2x + 1$ . Сколько существует программ, для которых при исходном числе 1 результатом является число 25 и при этом траектория вычислений не содержит число 21?

- 82) Исполнитель Июнь17 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Сделай нечётное

Выполняя первую команду, исполнитель увеличивает число на 1, а выполняя вторую – из числа  $x$  получает число  $2x + 1$ . Сколько существует программ, для которых при исходном числе 1 результатом является число 31 и при этом траектория вычислений не содержит число 25?

- 83) Исполнитель Июнь17 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2
3. Прибавить 3

Сколько существует программ, для которых при исходном числе 3 результатом является число 15 и при этом траектория вычислений не содержит число 8?

- 84) Исполнитель Июнь17 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2
3. Прибавить 4



Сколько существует программ, для которых при исходном числе 2 результатом является число 13 и при этом траектория вычислений не содержит число 6?

- 85) Исполнитель Июнь17 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 2 результатом является число 30 и при этом траектория вычислений содержит число 15?

- 86) Исполнитель Июнь17 преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 2 результатом является число 28 и при этом траектория вычислений содержит число 12?

- 87) Исполнитель K17 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

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

Программа для исполнителя K17 – это последовательность команд. Сколько существует таких программ, которые исходное число 3 преобразуют в число 13 и при этом траектория вычислений программы содержит число 9 и число 11?

- 88) Исполнитель K17 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

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

Программа для исполнителя K17 – это последовательность команд. Сколько существует таких программ, которые исходное число 1 преобразуют в число 13 и при этом траектория вычислений программы содержит число 4 и число 9?

- 89) Исполнитель R17 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

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

Программа для исполнителя R17 – это последовательность команд. Сколько существует таких программ, которые исходное число 1 преобразуют в число 12 и при этом траектория вычислений программы содержит число 7 и число 10?

- 90) Исполнитель R17 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

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

Программа для исполнителя R17 – это последовательность команд. Сколько существует таких программ, которые исходное число 2 преобразуют в число 15 и при этом траектория вычислений программы содержит число 4 и число 11?

- 91) Исполнитель R17 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

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

Программа для исполнителя R17 – это последовательность команд. Сколько существует таких программ, которые исходное число 2 преобразуют в число 14 и при этом траектория вычислений программы содержит число 6 и число 10?

- 92) Исполнитель R17 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

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

Программа для исполнителя R17 – это последовательность команд. Сколько существует таких программ, которые исходное число 3 преобразуют в число 20 и при этом траектория вычислений программы содержит число 9 и число 12?

- 93) (А.Е. Гребенкин, Екатеринбург) Исполнитель U18 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

1. Вычесть 1
2. Вычесть 3
3. Разделить нацело на 3

При выполнении команды 3 выполняется деление нацело (остаток отбрасывается). Программа для исполнителя U18 – это последовательность команд. Сколько существует таких программ, которые исходное число 22 преобразуют в число 2?

- 94) (А.Е. Гребенкин, Екатеринбург) Исполнитель U18 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

1. Вычесть 1
2. Вычесть 3
3. Взять остаток от деления на 4

Команда 3 выполняется только для чисел, больших, чем 4. Программа для исполнителя U18 – это последовательность команд. Сколько существует таких программ, которые исходное число 22 преобразуют в число 2?

- 95) Исполнитель Май18 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 3

Сколько существует программ, для которых при исходном числе 2 результатом является число 18 и при этом траектория вычислений содержит число 9 и не содержит число 14?

- 96) Исполнитель Май18 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 3

Сколько существует программ, для которых при исходном числе 2 результатом является число 20 и при этом траектория вычислений содержит число 10 и не содержит число 15?

- 97) Исполнитель Май18 преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 3

Сколько существует программ, для которых при исходном числе 3 результатом является число 20 и при этом траектория вычислений содержит число 15 и не содержит число 10?

- 98) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 3 результатом является число 46 и при этом траектория вычислений содержит число 12 и не содержит число 25?

- 99) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, для которых при исходном числе 5 результатом является число 52 и при этом траектория вычислений содержит число 15 и не содержит число 29?

- 100) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Умножить на 3
3. Умножить на 4

Сколько существует программ, для которых при исходном числе 2 результатом является число 60 и при этом траектория вычислений содержит число 16 и не содержит число 21?

- 101) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Умножить на 3
3. Умножить на 4

Сколько существует программ, для которых при исходном числе 2 результатом является число 70 и при этом траектория вычислений содержит число 8 и не содержит число 35?

- 102) (С.А. Скопинцева) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 4
3. Умножить на 4

Сколько существует программ, для которых при исходном числе 2 результатом является число 24 и при этом траектория вычислений содержит число 6 и не содержит число 8?

- 103) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, состоящих из 6 команд, для которых при исходном числе 1 результатом является число 20?

- 104) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2
3. Прибавить 3

Сколько существует программ, состоящих из 7 команд, для которых при исходном числе 3 результатом является число 22?

- 105) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1

2. Прибавить 3
3. Умножить на 2

Сколько существует программ, состоящих из 7 команд, для которых при исходном числе 2 результатом является число 25?

- 106) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ, состоящих из 8 команд, для которых при исходном числе 1 результатом является число 34?

- 107) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 4
3. Умножить на 2

Сколько существует программ, состоящих из 7 команд, для которых при исходном числе 3 результатом является число 27?

- 108) У исполнителя Калькулятор есть три команды, которым присвоены номера:

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

Сколько существует программ, которые преобразуют исходное число 1 в число 28, и при этом траектория вычислений содержит число 8 и не содержит чисел 10 и 11?

- 109) У исполнителя Калькулятор есть три команды, которым присвоены номера:

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

Сколько существует программ, которые преобразуют исходное число 2 в число 21, и при этом траектория вычислений содержит число 5 и не содержит чисел 8 и 11?

- 110) У исполнителя Калькулятор есть три команды, которым присвоены номера:

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

Сколько существует программ, которые преобразуют исходное число 3 в число 22, и при этом траектория вычислений содержит число 10 и не содержит чисел 8 и 15?

- 111) У исполнителя Калькулятор есть три команды, которым присвоены номера:

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

Сколько существует программ, которые преобразуют исходное число 4 в число 23, и при этом траектория вычислений содержит число 8 и не содержит чисел 11 и 18?

- 112) У исполнителя Калькулятор есть три команды, которым присвоены номера:

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

Сколько существует программ, которые преобразуют исходное число 5 в число 26, и при этом траектория вычислений содержит число 11 и не содержит чисел 13 и 15?

- 113) (С.С. Поляков) У исполнителя Калькулятор есть три команды, которым присвоены номера:

1. Прибавить 1

2. Прибавить 5
3. Умножить на 3

Найдите длину самой короткой программы, в результате выполнения которой при исходном числе 1 результатом является число 227.

- 114) (С.С. Поляков) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько существует программ минимальной длины, в результате выполнения которых при исходном числе 1 результатом является число 28?

- 115) (С.С. Поляков) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 5
3. Умножить на 3

Сколько существует программ минимальной длины, в результате выполнения которых при исходном числе 1 результатом является число 111?

- 116) (С.С. Поляков) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 5
3. Умножить на 3

Сколько разных чисел может быть получено из числа 1 с помощью программ, состоящих из 4 команд?

- 117) (С.С. Поляков) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 5
3. Умножить на 3

Сколько разных чисел может быть получено из числа 1 с помощью программ, состоящих из 7 команд?

- 118) (С.С. Поляков) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

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

Сколько разных чисел на отрезке [34, 59] может быть получено из числа 1 с помощью программ, состоящих из 6 команд?

- 119) (С.С. Поляков) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 5
3. Умножить на 3

Сколько разных чисел на отрезке [1000, 1024] может быть получено из числа 1 с помощью программ, состоящих из 8 команд?

- 120) (С.С. Поляков) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 5
3. Умножить на 3

Определите число, для получения которого из числа 1 существует 175 программ.

- 121) (А.Н. Носкин) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 2
2. Прибавить 5

Определите число, для получения которого из числа 5 существует 34 программы.

- 122) (А.Н. Носкин) Исполнитель Калькулятор преобразует число на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 2
2. Прибавить 4
3. Прибавить 5

Определите число, для получения которого из числа 31 существует 1001 программа.

- 123) (Е. Джобс) Исполнитель Остаточек преобразует числа и имеет следующие команды:

1. Прибавить 1
2. Умножить на 2
3. Прибавить остаток от деления на 4

Первая команда увеличивает число на единицу, вторая – увеличивает вдвое, третья команда добавляет к числу значение остатка от деления этого числа на 4. Определите, сколько существует чисел, из которых Остаточек может получить число 80 с помощью программы длиной не более 5 команд.

- 124) (Е. Джобс) Исполнитель Вычислитель преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 3
2. Умножить на 2

Первая команда увеличивает число на экране на 3, вторая – умножает на 2. Сколько существует программ, для которых при исходном числе 1 результатом является число 41 и при этом траектория вычислений содержит число 16 и не содержит числа 32?

- 125) (Е. Джобс) Исполнитель Вычислитель преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 2
2. Сделать простое

Первая команда увеличивает число на экране на 2, вторая – получает ближайшее большее простое число. Сколько существует программ, для которых при исходном числе 2 результатом является число 45 и при этом траектория вычислений содержит число 14 и не содержит числа 33?

- 126) (Е. Джобс) Исполнитель Вычислитель преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 3
2. Умножить на 2

Первая команда увеличивает число на экране на 3, вторая – увеличивает значение в два раза. Сколько существует программ, для которых при исходном числе 12 результатом является число 96?

- 127) (Е. Джобс) Исполнитель Умножитель преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Умножить на 2
2. Умножить на 3

Первая команда увеличивает число на экране в 2 раза, вторая – увеличивает значение в 3 раза. Сколько существует программ, для которых при исходном числе 8 результатом является число 3456, и при этом траектория вычислений содержит число 96.

- 128) (Е. Джобс) Исполнитель Простачок преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 2
2. Прибавить предыдущее
3. Прибавить следующее

Первая команда увеличивает число на 2, вторая – на предыдущее (например, число 5 будет преобразовано по правилу  $5 + 4$ ), третья – на следующее (аналогично, 5 по правилу  $5 + 6 = 11$ ). Сколько существует таких программ, которые исходное число 7 преобразуют в число 63, и при этом траектория вычислений не содержит число 43?

- 129) (Е. Джобс) Исполнитель ЛенивыйСчетовод преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 2
2. Прибавить 3
3. Дописать к числу справа 1

Первая команда увеличивает число на 2, вторая – на 3, третья – приписывает к текущему значению цифру 1 (например, для 10 результатом выполнения данной команды будет 101). Сколько существует таких программ, которые исходное число 3 преобразуют в число 25, при этом траектория вычислений содержит число 12?

- 130) (Е. Джобс) Исполнитель Простачок преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

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

Первая команда увеличивает число на 2, вторая – на 3, третья – увеличивает число вдвое. Сколько различных чисел может быть получено из числа 10 всеми возможными алгоритмами длиной 5 команд?

- 131) (А. Комков) Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Добавить справа 0
3. Добавить справа 1

Первая команда увеличивает число на 1. При выполнении второй команды, исполнитель справа к числу приписывает 0, а при выполнении третьей команды справа к числу приписывает 1. (например, для числа 10 результатом работы данных команд будут являться числа 100 и 101 соответственно).

Сколько существует программ, которые исходное двоичное число 100 преобразуют в двоичное число 11101?

- 132) (А. Комков) Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Добавить справа 0
3. Добавить справа 1

Первая команда увеличивает число на 1. При выполнении второй команды, исполнитель приписывает справа к числу 0, а при выполнении третьей команды приписывает справа к числу 1. (например, для числа 10 результатом работы данных команд будут являться числа 100 и 101 соответственно).

Сколько существует программ, которые исходное двоичное число 101 преобразуют в двоичное число 101110?

- 133) (А. Комков) Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть две команды, которым присвоены номера:

1. Вычесть 1
2. Убрать последнюю цифру справа



Первая команда уменьшает число на 1. Вторая команда убирает последнюю справа цифру, например, для числа 110 результатом работы данной команды будет являться число 11. Сколько существует программ, которые исходное двоичное число 10001 преобразуют в двоичное число 100?

- 134) **(А. Комков)** Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть две команды, которым присвоены номера:

1. **Вычесть 1**
2. **Убрать последнюю цифру справа**

Первая команда уменьшает число на 1. Вторая команда убирает последнюю справа цифру, например, для числа 110 результатом работы данной команды будет являться число 11. Сколько существует программ, которые исходное двоичное число 110111 преобразуют в двоичное число 110?

- 135) **(А. Комков)** Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть две команды, которым присвоены номера:

1. **Прибавить 1**
2. **Добавить слева 1**

Первая команда увеличивает число на 1. Вторая команда приписывает к двоичному числу слева 1, например, для числа 10 результатом работы данной команды будет являться число 110. Сколько существует программ, которые исходное двоичное число 100 преобразуют в двоичное число 110001?

- 136) **(А. Комков)** Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть две команды, которым присвоены номера:

1. **Прибавить 1**
2. **Добавить слева 1**

Первая команда увеличивает число на 1. Вторая команда приписывает к двоичному числу слева 1, например, для числа 10 результатом работы данной команды будет являться число 110. Сколько существует программ, которые исходное двоичное число 1 преобразуют в двоичное число 11111?

- 137) **(А. Комков)** Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть две команды, которым присвоены номера:

1. **Вычесть 1**
2. **Обнулить**

Первая команда уменьшает число на 1. Вторая команда обнуляет все ненулевые разряды, кроме старшего (например, для исходного числа 11101 результатом работы команды будет число 10000), если таких разрядов нет, то данная команда не выполняется. Сколько существует программ, которые исходное двоичное число 1100 преобразуют в двоичное число 100?

- 138) **(А. Комков)** Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть две команды, которым присвоены номера:

1. **Вычесть 1**
2. **Обнулить**

Первая команда уменьшает число на 1. Вторая команда обнуляет все ненулевые разряды, кроме старшего (например, для исходного числа 11101 результатом работы команды будет число 10000), если таких разрядов нет, то данная команда не выполняется. Сколько существует программ, которые исходное двоичное число 10001 преобразуют в двоичное число 1?

- 139) **(А. Комков)** Исполнитель Нолик преобразует двоичное число, записанное на экране. У исполнителя есть две команды, которым присвоены номера:

1. **Вычесть 1**
2. **Обнулить**

Первая команда уменьшает число на 1. Вторая команда обнуляет все ненулевые разряды, кроме старшего (например, для исходного числа 11101 результатом работы команды будет число 10000), если таких разрядов нет, то данная команда не выполняется. Сколько существует программ, которые исходное двоичное число 1000000 преобразуют в двоичное число 1000?



140) **(А. Комков)** Исполнитель Нолик преобразует число, записанное на экране в четверичной системе счисления. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 2
2. Прибавить 3
3. Добавить справа 0

Первая команда увеличивает число на 2. Вторая команда увеличивает число на 3. Третья команда приписывает к записи числа справа 0, например, для числа 123 результатом работы данной команды будет являться число 1230. Сколько существует программ, которые число 1, записанное в четверичной системе счисления, преобразуют в четверичную запись 100?

141) **(А. Комков)** Исполнитель Нолик преобразует число, записанное на экране в троичной системе счисления. У исполнителя есть две команды, которым присвоены номера:

1. Вычесть 2
2. Обнулить младший разряд

Первая команда уменьшает число на 2. Вторая команда обнуляет ненулевой младший разряд троичной записи числа. (Например, при выполнении этой команды число 21 преобразуется в число 20. Если в младшем разряде находится 0, то данная команда не выполняется). Сколько существует программ, которые троичное число 212, преобразуют в троичное число 10?