

05.12.

2/3 ~ 10
27280

Исполнитель РазДва 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 РазДва — это последовательность команд.

Сколько существует программ, которые преобразуют исходное число 2 в число 50, и при этом траектория вычислений содержит число 12 и не содержит числа 47?

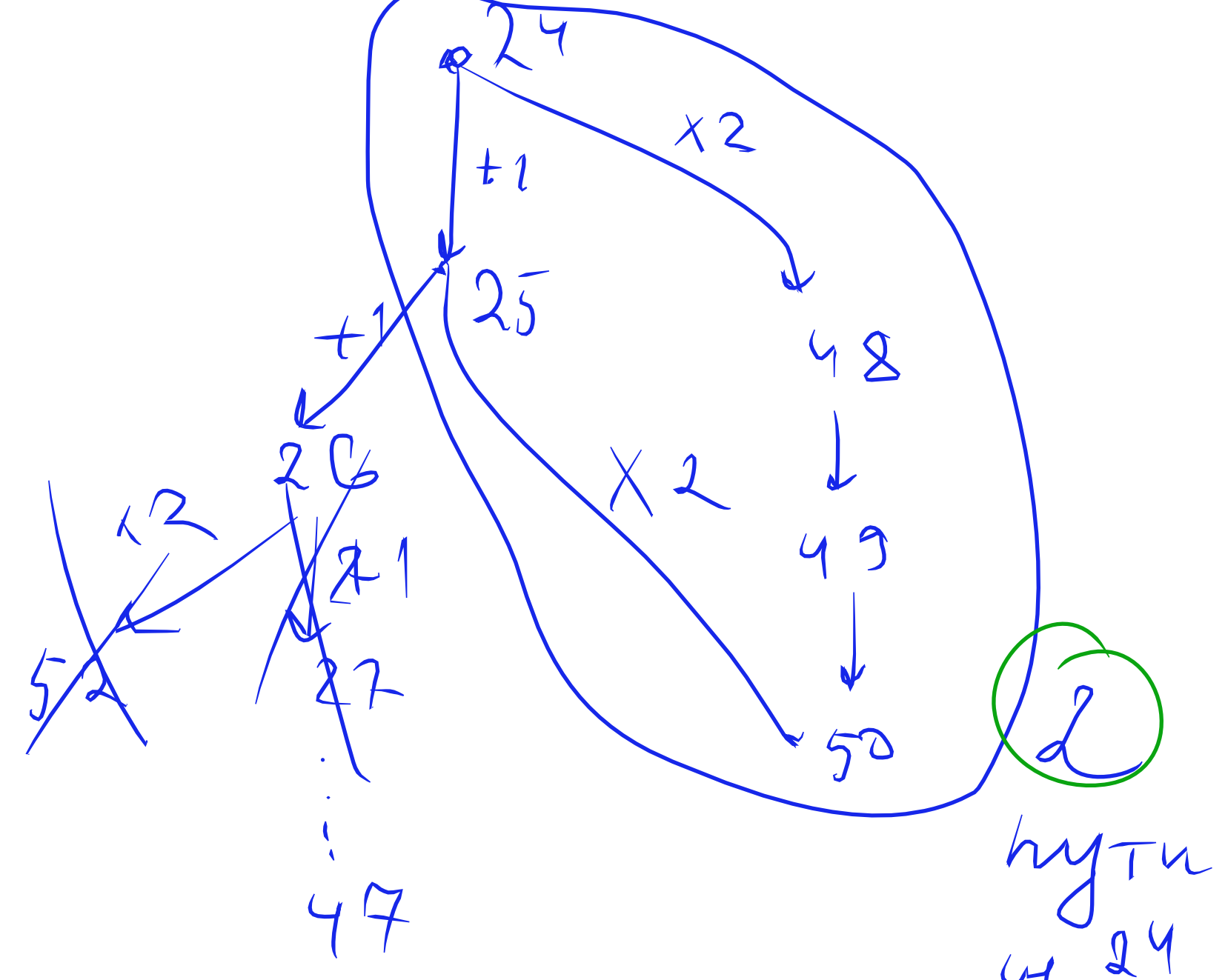
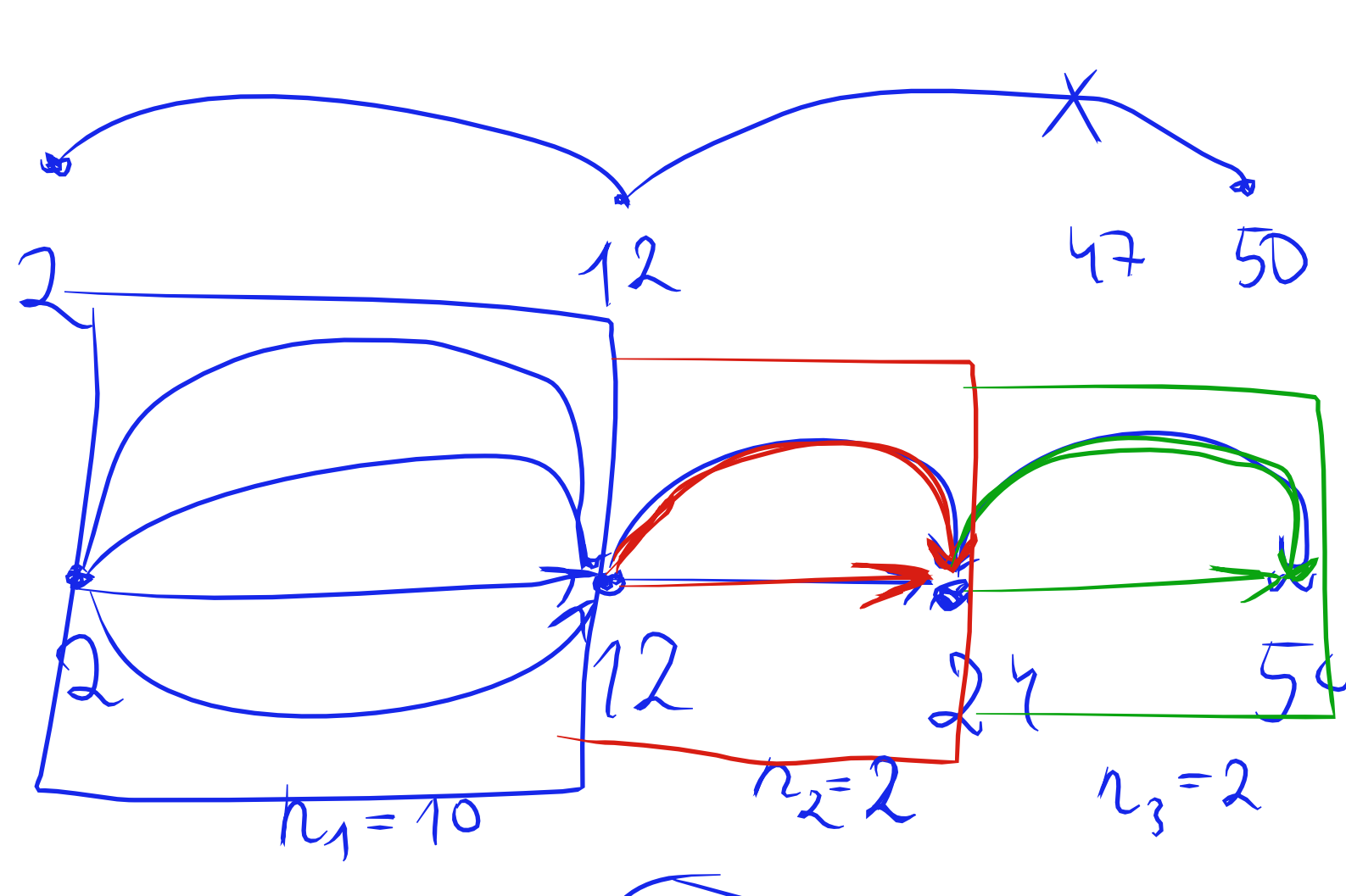
Траектория вычислений — это последовательность результатов выполнения всех команд программы. Например, для программы 212 при исходном числе 4 траектория будет состоять из чисел 8, 9, 18.

2	1	1
3	1	2
4	2	3 2
5	2	4
6	3	3 5
7	3	6
8	5	4 7
9	5	8
10	2	5 9
11	2	10
12	10	11 6
13	1	7 2
14	1	13
15	1	14
16	1	15

23	1	22
24	2	11 23
25	2	24
26	2	13 25

11.

46 2 23 45



$$10 \cdot 2 \cdot 2 = 40$$

2/3 программирование

~ 2 (56)

$$56 : 4 \Rightarrow 256 : 4$$

1) число n и N последних цифр

$X = 123456$

$N = 3$

$X \% 10 \neq N$

2) число n и N первых цифр

— определить как-во цифр в числе K ($K=6$)

— разделить число на $10 \neq (K-1-(N+1)) = 10 \neq (K-N)$

~ 4. 2 задачи:

$\max \left\{ \begin{array}{l} \text{sum} \\ \text{diff} \end{array} \right\}$

~ 3. 1 2 3 2 1

```
i = 0; sum_even = 0; sum_odd = 0
while N > 0:
    if i % 2 == 0:
        sum_even += N % 10
    else:
        sum_odd += N % 10
    i += 1
    N //= 10
```

Проверка на делимость. Определение делимости и их сумма. Определение "простоты" числа

1. Проверить, что $N:M$:

$$N \% M == 0$$

2. Определение делителей числа X

```
deliteli = []
for i in range(1, X+1):
    if X % i == 0:
        deliteli.append(i)
k = 0
for i in range(1, X+1):
    if X % i == 0:
        k += 1
```

3. Определение "простоты" числа

I. $[1; x]$, if $k == 2$: II. $[2; x-1]$, if $k == 0$:

тогда
число
 X
простое

III. is-simple = True
for i in range(2, X):
 if X % i == 0:
 is-simple = False
 break

Если X разделилось на какое-то число i , то оно уже ТОЧНО не простое и дальше проверять нет смысла

4.*

Можно определять простоту числа, проходя до корня числа, потому что если число X разделилось на какое-то число $Y < \sqrt{X}$, то тогда оно разделилось и на число $X // Y > \sqrt{X}$.
Можно определять делители числа, проходя до корня числа, тогда при каждой итерации надо записывать в deliteli два числа: $i, X // i$.
Можно определять количество делителей числа, проходя до корня числа, тогда при каждой итерации надо прибавлять: 2, если $i \neq X // i$, 1, если $i = X // i = \sqrt{X}$

Корень целого числа

square root

I. from math import sqrt
sqrt(5) # 2.236

II. 5 ** 0.5 # 2.236
Float

2. from math import ceil
ceil(sqrt(5)) # 3

8. range

Created with IDroo.com