Задача 1. Таблица квадратов

В задаче требуется записать формулу для вычисления числа, расположенного в строке с номером i и столбце с номером j.

В этой ячейке находится квадрат числа $10 * i + j$

Получаем формулу $(10 * i + j) * (10 * i + j)$

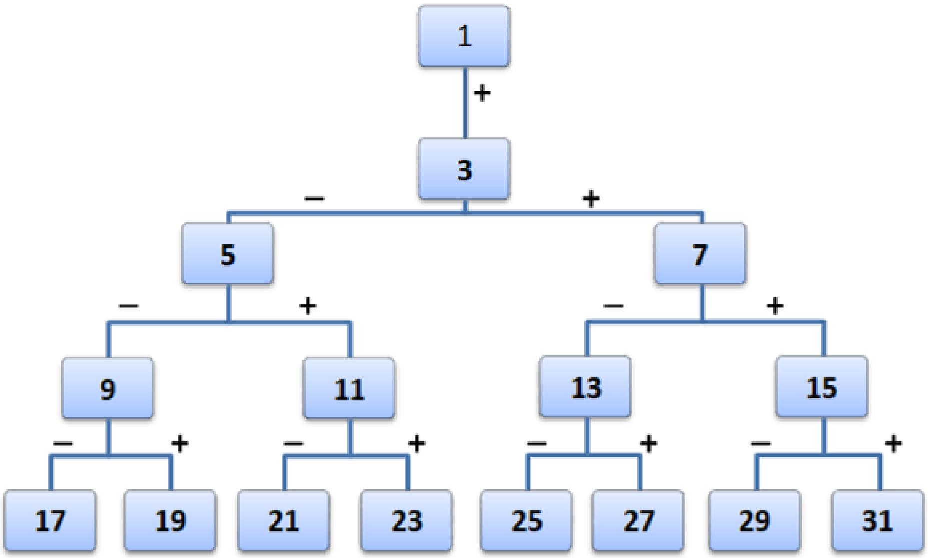
Задача 2. Калькулятор

В задаче с числом x можно выполнить 2 операции:

$x \times 2 + 1$ и $x \times 2 - 1$

Необходимо с помощью этих операций из числа 1 получить 25 и 725.

Для ответа на первый вопрос построим дерево вариантов выполнения операций.



Заметим, что дерево содержит все нечетные числа по одному разу. То есть с помощью заданных операций можно получать только нечетные числа, и для каждого нечетного числа есть только один способ его получения.

Выпишем по дереву ответ для числа 25:

$+ + - -$

Для ответа на второй вопрос будем делать операции с конца.

Посмотрим, как могло получиться 725. Если последней была операция $+$, то предыдущее число было 362 ($362 \times 2 + 1 = 725$). А если последней операцией была операция –, то предыдущее число было 363 ($363 \times 2 - 1 = 725$). Но мы уже выяснили, что в нашей последовательности выполнения операций могут быть только нечетные числа, поэтому предыдущим числом должно было быть 363, а последняя операция –

Аналогично рассмотрим 2 способа получить число 363: $363 = 181 \times 2 + 1$ и $363 = 182 \times 2 - 1$ Выберем число 181, потому что оно нечетное и предпоследняя операция $+$

Продолжая такие же рассуждения дальше получим:

$181 = 91 \times 2 - 1$

$91 = 45 \times 2 + 1$

$45 = 23 \times 2 - 1$

$23 = 11 \times 2 + 1$

$11 = 5 \times 2 + 1$

$5 = 3 \times 2 - 1$

$3 = 1 \times 2 + 1$

Запишем эти операции в правильном порядке, получим:

$+ - + + - + - + -$

Задача 3. Строки

Для ответа на первый вопрос заметим, что строка с номером k заканчивается на k цифр k. Поэтому девятая строка будет заканчиваться на 9 девяток. Перед ними будет записан номер предыдущей строки 8.

Определим длину последней строки. Для получения длины каждой следующей строки надо длину предыдущей строки увеличить в 2 раза (так как она записывается 2 раза), а потом прибавить 1, так как приписывается ещё номер текущей строки. Получаем такие длины всех строк:

1
3
7
15
31
63
127
255
511

Таким образом, длина последней строки 511. Можно было не вычислять длины всех строк, а заметить, что длина строки с номером k равна $2^k - 1$.

Для ответа на третий вопрос заметим, что количество цифр 2 в каждой строке больше, чем в предыдущей строке на 1. Новая двойка появляется, как увеличенная 1, переписанная из предыдущей строки. Поэтому в последней строке будет 9 цифр 2.

Для ответа на четвертый вопрос можно сложить длины всех строк, полученные при ответе на второй вопрос. Получится 1013.

Для ответа на пятый вопрос рассмотрим девятую строку последовательности. В ней сначала записаны 255 цифр из восьмой строки, потом 255 цифр – увеличенные цифры восьмой строки. И в конце число 9. Получается, что цифра, расположенная на 200-м месте – это цифра, расположенная в восьмой строке на 200-м месте.

Рассмотрим восьмую строку. Она состоит из 127 цифр седьмой строки, 127 увеличенных цифр седьмой строки. и числа 8. Получается, что цифра, расположенная в восьмой строке на 200-м месте, – это увеличенная цифра седьмой строки, расположенная в седьмой строке на 73 месте $(200 - 127)$.

Рассмотрим седьмую строку. Она состоит из 63 цифр шестой строки, 63 увеличенных цифр шестой строки, и числа 7. Тогда цифра, расположенная на 73 месте седьмой строки, – это увеличенная цифра шестой строки, расположенная в шестой строке на 10-м месте $(73 - 63)$.

Десятую цифру строки можно получить, выписывая цифры последовательности по заданным в условии задачи правилам:

1222333233

Получаем цифру 3. Поскольку она должна быть увеличена 2 раза: при переходе от шестой строки к седьмой, и при переходе от седьмой строки в восьмую, то получается, что на 200-м месте располагается цифра 5.

Запишем все ответы на вопросы:

1. 8999999999
2. 511
3. 9
4. 1013
5. 5

Задача 4. Карточки

Посмотрим на первую таблицу.

В первой строке уже есть буквы A и B, поэтому в первой ячейке должна быть C. Также уже есть цифры 3 и 2, поэтому в первой ячейке должна быть цифра 1.

Во втором столбце не хватает буквы C и цифры 2, запишем $C2$.

В оставшихся ячейках сначала расставим буквы. Поскольку во второй строке уже есть буква C, а в третьем столбце уже есть буква B, то на их пересечении должна быть буква A. Тогда в первой ячейке второй строки поставим B. Аналогично, в первой ячейки третьей строки запишем A, а в последней ячейке C.

Теперь расставим цифры:

Поскольку $B1$ и $B2$ уже есть, то в первой ячейке второй строки к букве B припишем 3. Тогда в первом столбце не хватает цифры 2, припишем её к A, а во второй строке не хватает цифры 1, припишем её к A. Остается приписать 3 к последней букве C.

Получаем такое заполнение таблицы:

$C1$	$A3$	$B2$
$B3$	$C2$	$A1$
$A2$	$B1$	$C3$

Посмотрим на вторую таблицу, здесь нужно использовать 4 буквы и 4 цифры.

Заполним третью строку. Посмотрим ячейку на пересечении третьей строки и второго столбца. Поскольку в третьей строке уже есть буквы B и C, а во втором столбце есть буква A, то здесь надо поставить D. Тогда в следующей ячейке надо поставить букву A. Поскольку в третьей строке уже есть цифры 3 и 4, а в третьем столбце есть цифра 1, то на их пересечении надо поставить цифру 2. Тогда у буквы D должна быть цифра 1.

Заполним теперь второй столбец. Во втором столбце уже есть буквы A и D, а в первой строке есть буква C, поэтому на их пересечении поставим букву B. Соответственно, оставшаяся буква во втором столбце - B. Теперь надо расставить этим буквам цифры. Поскольку 3 и 1 уже есть во втором столбце, то надо расставить 2 и 4. Но $B4$ уже есть в таблице, значит к B припишем 2, а к C - 4.

Заполним первую строку. В ней уже есть цифры 1 и 2, осталось записать 3 и 4. Но в первом столбце уже есть 3, а в четвертом уже есть 4, поэтому на пересечении первой строки и первого столбца запишем 4, а на пересечении первой строки и последнего столбца запишем 3. Теперь добавим в эти ячейки буквы. В первой строке не хватает букв A и D. Но пара $A3$ уже есть в таблице, поэтому к цифре 3 припишем D, а к цифре 4 - A.

Аналогично заполним последний и первый столбцы.

В результате получим таблицу

$A4$	$B2$	$C1$	$D3$
$B1$	$A3$	$D4$	$C2$
$C3$	$D1$	$A2$	$B4$
$D2$	$C4$	$B4$	$A1$

Задача 5. Время в школе

Для того, чтобы определить общую длительность всех уроков, нужно количество уроков n умножить на длительность урока a.

Между n уроками будет $(n-1)$ перемена, из них обычных перемен будет $(n-2)$, поэтому общая длительность перемен находится по формуле $(n-2)*b$.

Для получения общего времени, которое Вася провел в школе, остается добавить длительность большой перемены 30.

Описанные рассуждения можно записать в виде следующего кола на языке программирования Python.

```
n = int(input())
a = int(input())
b = int(input())
print(n * a + (n - 2) * b + 30)
```

Задача 6. Разноэтажный дом

Для начала рассмотрим расположение квартир в первом подъезде. На первом этаже расположены квартиры с номерами 1, 2, 3; на втором этаже расположенные квартиры с номерами 4, 5, 6 и так далее. Получим формулы для нахождения номеров квартир первого подъезда:

- $(K-1)*3+1$;
- $(K-1)*3+2$;
- $(K-1)*3+3$.

где K — номер этажа.

Для второго и третьего подъезда формулы изменятся незначительно. А именно, необходимо добавить в формулы ещё одно слагаемое — количество квартир в подъездах с меньшими номерами (в первом подъезде для формул второго подъезда, в первых двух подъездах для третьего подъезда). Рассмотрим решение, использующее полученные формулы:

```
FLATS = 3
a = int(input())
b = int(input())
c = int(input())
k = int(input())

prev = 0
print((k - 1) * FLATS + 1)
print((k - 1) * FLATS + 2)
print((k - 1) * FLATS + 3)

prev += FLATS * a
print(prev + (k - 1) * FLATS + 1)
print(prev + (k - 1) * FLATS + 2)
print(prev + (k - 1) * FLATS + 3)

prev += FLATS * b
print(prev + (k - 1) * FLATS + 1)
print(prev + (k - 1) * FLATS + 2)
print(prev + (k - 1) * FLATS + 3)
```

Переменная FLATS хранит в себе количество квартир на площадке, переменная prev — количество квартир в подъездах с меньшими номерами.

Однако данное решение наберёт не более 40 баллов, ибо не учитывает, что в некоторых подъездах может не быть этажа с номером K. Для получения полного балла достаточно добавить следующую модификацию в решение — перед выводом номеров квартир проверять, что в данном подъезде достаточное количество этажей.

Итоговое решение выглядит так:

```
FLATS = 3
a = int(input())
b = int(input())
c = int(input())
k = int(input())

prev = 0
if a >= k:
    print((k - 1) * FLATS + 1)
    print((k - 1) * FLATS + 2)
```

```
    print((k - 1) * FLATS + 3)

prev += FLATS * a
if b >= k:
    print(prev + (k - 1) * FLATS + 1)
    print(prev + (k - 1) * FLATS + 2)
    print(prev + (k - 1) * FLATS + 3)

prev += FLATS * b
if c >= k:
    print(prev + (k - 1) * FLATS + 1)
    print(prev + (k - 1) * FLATS + 2)
    print(prev + (k - 1) * FLATS + 3)
```

Задача 7. Длина числа

Если немного перефразировать условие, в задаче требуется посчитать суммарное количество цифр во всех целых числах от L до R. Можно сделать это, пробежавшись от L до R циклом, на каждом шаге добавляя к ответу количество цифр в очередном числе:

```
L = int(input())
R = int(input())
sum = 0
for i in range(L, R+1):
    sum += len(str(i))
print(sum)
```

Такой способ набирает 44 балла, однако при полных ограничениях не проходит по времени. Например, на третьем тесте из условия приходится перебирать циклом уже 10^9 чисел, а в худшем случае это количество может вообще достигать 10^{17}.

Для решения на полный балл применим следующий прием — вместо вычисления количества от L до R вычислим это количество во всех числах от 1 до R и вычтем из него количество в числах от 1 до $L-1$. Тем самым мы свели нашу задачу к двум однотипным задачам про числа от 1 до n, в которых есть только правая граница, а не обе границы сразу.

Как решать задачу про суммарное количество цифр во всех целых числах от 1 до n? Найдем k — количество цифр в числе n. Тогда наш диапазон попадут все однозначные, двузначные $\ldots$ $k-1$-значные числа. Пробежимся циклом по всем этим длинам (это не больше 17 итераций) и прибавим к ответу количество всех чисел такой длины (равное $9 \cdot 10^{длина-1}$), умноженное на эту длину. А чисел длины k в наш диапазон попадет $n - 10^{k-1} + 1$.

```
def from_1_to_n(n):
    k = len(str(n))
    sum = (n - 10**(k-1) + 1) * k
    for i in range(1, k):
        sum += 9 * 10**(i-1) * i
    return sum

L = int(input())
R = int(input())
print(from_1_to_n(R) - from_1_to_n(L-1))
```

Отметим, что первую подгруппу на 20 баллов ($R \leqslant 50$) можно было решить без использования циклов с помощью нескольких условных операторов. Например, разобрав отдельно три случая: обе границы — однозначные числа (ответ $R-L+1$), обе границы — двузначные числа (ответ $2\cdot(R-L+1)$) и, наконец, левая граница — однозначная, а правая — двузначная (ответ $2 \cdot (R-9) + (10-L)$).