

6. Волшебный ВФ (mbf)

60 очков

В школе Петя узнал, что во многих восточноазиатских языках текст может записываться как слева направо, так и сверху вниз. Ему стало интересно, можно ли написать текст так, чтобы его можно было одновременно прочитать обоими способами. Петя не избавился от своей идеи и на уроке программирования, и хочет теперь написать код так, чтобы он решал задачи как при чтении его по строкам (сверху вниз и слева направо), так и при чтении по столбцам (слева направо и сверху вниз). Конечно, для этого был нужен подходящий язык, и Петя выбрал для своего эксперимента язык, называющийся ВФ.

Памятью программы на ВФ является массив M бесконечной длины, ячейки которого пронумерованы слева направо (M_0, M_1, \dots). В каждой ячейке может быть сколь угодно большое неотрицательное целое число. Также в программе есть стрелка, которая при запуске кода указывает на самую левую ячейку массива (M_0).

Выполнение программы начинается с её первой команды и после выполнения каждой команды в общем случае переходит к следующей команде. Всего в этом языке шесть команд, каждая из которых обозначается одним знаком:

Команда	Значение
>	Передвигает стрелку на одну ячейку вправо.
<	Если стрелка не указывает на самую левую ячейку, то передвигает её на одну ячейку влево; в противном случае не делает ничего.
+	Увеличивает содержание указанной ячейки на один.
-	Если содержание указанной ячейки положительно, то уменьшает его на один; в противном случае не делает ничего.
[Если в указанной ячейке ноль, то прыгает по коду программы вперёд до соответствующего знака ']' ; в противном случае не делает ничего.
]	Если в указанной ячейке не ноль, то прыгает по коду программы назад до соответствующего знака '[' ; в противном случае не делает ничего.

Помогите Пете написать волшебные программы на ВФ для решения пяти описанных ниже задач.

Как решение каждого подзадания нужно предоставить программный код ВФ, отформатированный в виде квадрата $N \times N$ (где $1 \leq N \leq 1000$). Код может отличаться при чтении по рядам и по столбцам, но должен правильно решать заданную задачу при чтении каждым способом. Текст решения может содержать только знаки '>', '<', '+', '-', '[' и ']' и должен полностью заполнять квадрат. Ни при каких разрешённых входных данных программа не должна затратить на свою работу более 10 миллионов выполнений команд.

Пример. Задача: Сложить значения ячеек M_0 и M_1 и записать результат в ячейке M_0 . Во всех остальных ячейках массива при запуске кода записано значение 0.

```
>[-<+>]
[++++<>
<----><
Решение: +++++<<
>---]><
-<><>[>
]><<<<<
```


максимальное число из присутствующих в массиве в момент запуска программы и записать его в ячейку M_0 . Можно учитывать, что ответ не больше 50, и что не более 100 ячеек имеют большее нуля значение. Все остальные ячейки кроме M_0 могут в конце программы содержать любые числа.

Пример входных данных:

0	23	50	31	32	12	0	0	...
---	----	----	----	----	----	---	---	-----

Пример выходных данных:

50	0	0	0	...
----	---	---	---	-----