

## Задача 6. Стековый калькулятор

Имя входного файла:	stack.in
Имя выходного файла:	stack.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт
Максимальная оценка:	100 баллов

В кабинете информатики есть старый *стековый калькулятор*. Он содержит  $K$  ячеек памяти, организованных в виде *стека*. Первая ячейка называется *вершиной* стека. На индикаторе калькулятора отображается содержимое вершины стека, если стек не пуст.

Над стеком может выполняться операция *проталкивания*. Применение этой операции приводит к записи числа на вершину стека. Перед этим, если в стеке уже были числа, то каждое из них перемещается в ячейку с номером на единицу больше. Если в стеке уже находится  $K$  чисел, то выполнение операции проталкивания невозможно.

Калькулятор позволяет выполнять арифметические операции. Они применяются к числам, хранящимся во второй и первой ячейках стека. Результат операции записывается в первую ячейку стека, а число из второй ячейки удаляется. После этого, если третья ячейка не пуста, то число из неё переписывается во вторую, если есть число в четвертой ячейке — оно переписывается в третью и так далее до последней занятой ячейки, которая становится пустой. Для выполнения арифметической операции в стеке должно быть хотя бы два числа. Например, при выполнении операций сложения или умножения, значения соответственно суммы или произведения чисел из первой и второй ячеек помещаются на вершину стека. Операция вычитания выполняется так: из содержимого второй ячейки стека вычитается содержимое первой ячейки.

Известно, что калькулятор неисправен. Из цифровых клавиш работает только клавиша «1». Нажатие этой клавиши приводит к проталкиванию в стек числа 1. Например, попытка набрать число 11, два раза нажав клавишу «1», приводит к тому, что в стек два раза проталкивается число 1. Из операций работают только сложение (клавиша «+»), умножение (клавиша «\*») и вычитание (клавиша «-»). Если в результате вычитания получено отрицательное число, то калькулятор зависает.

Легко заметить, что на индикаторе возможно получить произвольное натуральное число. Например, чтобы получить число 3, необходимо дважды нажать клавишу «1», затем клавишу «+» (на индикаторе после этого появится число 2), затем один раз нажать клавишу «1» и один раз — клавишу «+». При  $K > 2$  того же результата можно достичь иначе, трижды нажав клавишу «1», а затем два раза клавишу «+». Дополнительно используя операции умножения и вычитания, в некоторых случаях число  $N$  можно получить быстрее, чем сложив  $N$  единиц.

Требуется написать программу, которая определяет, каким образом можно получить на индикаторе калькулятора заданное число  $N$ , выполнив минимальное количество нажатий клавиш. Стек изначально пуст.

### Формат входных данных

В единственной строке входного файла записаны два натуральных числа —  $N$  и  $K$  ( $1 \leq N \leq 10^9$ ,  $2 \leq K \leq 100$ ).

### Формат выходных данных

В первой строке выходного файла выведите минимальное количество нажатий клавиш, необходимых для получения числа  $N$ . Если число нажатий не превосходит 200, то дополнительно выведите во второй строке оптимальную последовательность нажатий, приводящих к появлению данного числа на индикаторе.

Последовательность необходимо выводить без пробелов. Клавиши обозначаются символами «1» — протолкнуть число 1 в стек, «+» — выполнить операцию сложения, «\*» —

---

выполнить операцию умножения, «-» — вычесть из значения второй ячейки стека значение первой ячейки.

В результате выполнения выведенной последовательности нажатий на индикаторе должно отображаться число  $N$ . Если оптимальных последовательностей нажатий несколько, разрешается выводить любую из них.

**Примечания**

Решения, корректно работающие при  $N \leq 100$  и  $K \leq 10$ , будут оцениваться из 40 баллов.

Решения, корректно работающие при  $N \leq 10^6$  и  $K \leq 100$ , будут оцениваться из 80 баллов.

**Примеры входных и выходных данных**

<b>stack.in</b>	<b>stack.out</b>
6 3	9 111++11+*
11 4	15 11+111+++*11+*1-