

Задача 1. Большое число

Имя входного файла: `bignum.in`
Имя выходного файла: `bignum.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дано целое число N , состоящее из четного количества десятичных цифр. Над ним последовательно производятся следующие действия:

1. цифры числа разделяются на две равные половины;
2. левая и правая половины разворачиваются, то есть порядок следования цифр меняется на противоположный;
3. аналогичные действия выполняются для частей числа без первой и последней цифры, и так далее.

Когда останется последняя цифра первой половины числа и первая — второй, процесс останавливается, так как разворачивать их не имеет смысла.

Рассмотрим пример. Пусть $N = 1234567890$. Тогда в процессе выполнения указанных действий будет получена следующая цепочка: 5432109876, 5123478906, 5143298706, 5142389706. Ваша задача — узнать результат последовательности указанных преобразований.

Формат входного файла

Входной файл содержит единственное число N (не более 10000 цифр). Допускаются нули в начале записи числа.

Формат выходного файла

Выходной файл должен содержать единственное число — результат применения всех действий.

Пример входных и выходных файлов

<code>bignum.in</code>	<code>bignum.out</code>
1234567890	5142389706
000123	000231
012039	201390

Задача 2. Ох, уж эти скобки

Имя входного файла: brackets.in
Имя выходного файла: brackets.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Математическое выражение записано в виде произведения:

$$(\pm a_2 x^2 \pm a_1 x \pm a_0) \cdot (\pm b_2 x^2 \pm b_1 x \pm b_0) \cdot (\pm c_2 x^2 \pm c_1 x \pm c_0) \dots$$

Внутри каждой из N скобок произведения находится выражение вида:

$$\pm a_2 x^2 \pm a_1 x \pm a_0,$$

где хотя бы один из коэффициентов a_i не равен нулю (b_i , c_i и т. д., аналогично). Требуется составить программу, которая перемножает выражения в скобках и выводит полученную функцию в виде многочлена с приведенными по степеням x слагаемыми, то есть в виде:

$$\pm q_{2N} x^{2N} \pm q_{2N-1} x^{2N-1} \pm \dots \pm q_3 x^3 \pm q_2 x^2 \pm q_1 x \pm q_0.$$

Формат входного файла

В первой строке входного файла находится число N ($1 \leq N \leq 6$).

Во второй строке находится выражение из N пар скобок. Внутри каждой пары скобок находится выражение в виде « $\pm a_2 x^2 \pm a_1 x \pm a_0$ », где « \pm » — это или знак «+», или знак «-». Значение каждого из коэффициентов a_i , b_i , c_i и т.д. не превышает 10. Выражение составлено по следующим правилам:

1. Всё выражение записывается, начиная со старшей степени переменной по убыванию степеней.
2. Если какой-то коэффициент равен нулю, то этот коэффициент и соответствующий ему x опускаются в записи вместе с арифметическим знаком. Исключением является случай, когда все коэффициенты равны нулю. В этом случае вместо всего выражения указывается единственный коэффициент 0.
3. Если $a_i = \pm 1$ и $i > 0$, то единица перед соответствующим ему x не ставится.
4. Если первый отличный от нуля коэффициент положителен, то знак «+» перед ним опускается.
5. В выражении отсутствуют пробельные символы (пробел, табуляция) и знаки умножения.

Формат выходного файла

В первой строке выходного файла выведите результат раскрытия скобок в исходном выражении в следующем формате:

$$\pm q_{2N} x^{2N} \pm q_{2N-1} x^{2N-1} \pm \dots \pm q_1 x \pm q_0.$$

Формат выражения должен полностью соответствовать описанию для входного файла. Скобки вокруг степеней ставить не нужно, они приведены здесь только для читабельности.

Пример входных и выходных файлов

brackets.in	brackets.out
1 (3x^2+2x-1)	3x^2+2x-1
2 (4x^2+3x+5) (2x^2+4x+1)	8x^4+22x^3+26x^2+23x+5
3 (-x+7) (x) (x^2+x+1)	x^4+6x^3+6x^2+7x
6 (1) (2) (3) (4) (5) (6)	720

Задача 3. Разбиение числа

Имя входного файла: `decomp.in`
Имя выходного файла: `decomp.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Факториалом числа n называется произведение $n! = 1 \cdot 2 \cdot \dots \cdot n$ при $n > 0$ и $n! = 1$ при $n = 0$.

Количество сочетаний из n элементов по k определяется следующим образом:

$$\binom{n}{k} = \begin{cases} \frac{n!}{k!(n-k)!}, & 0 \leq k \leq n; \\ 0, & k > n. \end{cases}$$

В математике такие числа называются также биномиальными коэффициентами. Требуется представить заданное число P в виде суммы трех биномиальных коэффициентов:

$$P = \binom{a}{1} + \binom{b}{2} + \binom{c}{3}, 0 \leq a < b < c.$$

Формат входного файла

Входной файл содержит единственное число P ($1 \leq P \leq 10^{18}$).

Формат выходного файла

В выходной файл выведите искомые числа a, b, c ($0 \leq a < b < c$), разделённые пробелами. Если задача не имеет решения, выведите три нуля.

Пример входных и выходных файлов

<code>decomp.in</code>	<code>decomp.out</code>
42	1 4 7
31	1 5 6

Задача 4. НОК

Имя входного файла: `lcm.in`
Имя выходного файла: `lcm.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Наименьшим общим кратным (НОК) нескольких чисел называется наименьшее натуральное число, которое делится на каждое из этих чисел. Заданы два числа N и K . Требуется найти набор из N различных натуральных чисел, наименьшее общее кратное которых равняется K . Среди всех этих чисел не должно быть единицы и самого числа K .

Формат входного файла

В первой строке входного файла записаны через пробел два числа N и K ($1 \leq N \leq 1000$, $1 \leq K \leq 10^9$).

Формат выходного файла

В выходной файл в первой строке выведите искомый набор из N чисел, разделённых пробелами. Если Вы смогли найти несколько наборов, то выведите любой из них.

Если требуемого набора не существует, тогда выведите -1.

Пример входных и выходных файлов

<code>lcm.in</code>	<code>lcm.out</code>
2 14	2 7
12 20736	3 9 27 81 256 128 64 32 16 8 4 2
17 42	-1
7 123456	2 3 4 6 30864 41152 61728