

Задача А. Транзисторы над Пекином

Имя входного файла: `transistor.in`
Имя выходного файла: `transistor.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Всемирно известный профессор В.В. Адимов продолжает свои разнообразные исследование устойчивости транзисторов. Теперь в голову ему пришла следующая задача: в доме N этажей, профессор хочет выяснить номер максимального этажа, падение с которого оставляет транзистор целым. Поскольку профессор исследует сферичиские транзисторы в вакууме, то можете считать что разбившись при падении с этажа f транзистор обязательно разобьется при падении с этажа $f + 1$. Дополнительно поставлено условие, что разрешено проведение не более чем K испытаний.

Эта задача была поручена именно вам, как самому успешному аспиранту профессора Адимова. Поскольку транзисторы нынче в цене, но наука все-таки дороже, то необходимо выяснить, какое минимальное количество транзисторов необходимо закупить, чтобы успешно провести эксперимент даже если вам будет катастрофически не везти.

Формат входных данных

В первой и единственной строке входного файла содержатся два целых положительных числа N и K не превосходящих 2000.

Формат выходных данных

Выведите единственное число - ответ на поставленную задачу. Если для данных N и K возможна ситуация, при которой мы не сможем получить ответ на вопрос даже имея неограниченный запас бесплатных транзисторов выведите -1 .

Примеры

<code>transistor.in</code>	<code>transistor.out</code>
4 2	-1
4 3	2

Задача В. Письма

Имя входного файла: `letter.in`
Имя выходного файла: `letter.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Расставаясь на летние каникулы, школьники Оксана и Петя договорились писать друг другу письма. Петя не отличается большой фантазией, поэтому каждое его письмо получается следующим образом. Вначале он задумывает n строк $\alpha_1, \dots, \alpha_n$ ($1 \leq n \leq 14$). Каждое письмо Пети представляет собой результат конкатенации (склеивания) этих строк, взятых в некотором порядке.

Одновременно, Петя отличается симпатичностью, поэтому его первое письмо представляет собой лексикографически минимальную возможную строку, получаемую описанным выше образом. Второе его письмо будет второй в лексикографическом порядке такой строкой. В общем случае k -е письмо Пети — это k -я в лексикографическом порядке строка. (В случае, если одна и та же строка может получаться конкатенацией $\alpha_1, \dots, \alpha_n$ несколькими способами, то она учитывается соответствующее число раз.)

Оксана не уверена, что у Пети хватит терпения писать письма все каникулы, поэтому она хочет заранее знать, каковым будет k -е по счету письмо (письма нумеруются с 1). Помогите ей решить эту задачу!

Формат входных данных

В первой строке входного потока заданы числа n и k ($1 \leq k \leq n!$). В последующих n строках входного потока заданы строки $\alpha_1, \dots, \alpha_n$. Каждая строка представляет собой непустую последовательность из строчных латинских букв (без пробелов) длины не более 20.

Формат выходных данных

Выведите в первой и единственной строке искомое k -е письмо Пети.

Примеры

<code>letter.in</code>	<code>letter.out</code>
3 2 aa bb cc	aaccbb

Задача С. Представление числа

Имя входного файла: `number.in`
Имя выходного файла: `number.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Учительница математики попросила школьников составить арифметическое выражение, так чтобы его значение было равно данному числу N , и записать его в тетради. В выражении могут быть использованы натуральные числа, не превосходящие K , операции сложения и умножения, а также скобки. Петя очень не любит писать, и хочет придумать выражение, содержащее как можно меньше символов. Напишите программу, которая поможет ему в этом.

Формат входных данных

В первой строке входного файла записаны два натуральных числа: N ($1 \leq N \leq 10000$) — значение выражения и K ($1 \leq K \leq 10000$) - наибольшее число, которое разрешается использовать в выражении.

Формат выходных данных

В единственной строке выходного файла выведите выражение с данным значением, записывающееся наименьшим возможным количеством символов. Если решений несколько, выведите любое из них.

Примеры

<code>number.in</code>	<code>number.out</code>
7 3	3+3+1
15 20	15
96 1	(1+1+1+1)*(1+1+1+1)*(1+1+1)*(1+1)

Замечание

При подсчете длины выражения учитываются все символы: цифры, знаки операций, скобки.

Задача D. Кубики

Имя входного файла: `abbaa.in`
Имя выходного файла: `abbaa.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Серёжа — обычный мальчик. Он ещё маленький, поэтому пока не умеет читать. И вот, однажды, родители подарили ему набор параллелепипедов размером $2 \times 1 \times 1$ с буквами на гранях каждого единичного кубика. Однако, Серёжа ещё очень маленький, и родителям не хочется шокировать его огромным количеством разных букв. Поэтому, на кубиках написаны только первые две буквы алфавита — А и В. На каждом параллелепипеде написаны обе эти буквы, по одной букве на всех гранях каждого единичного кубике.

Обрадовавшись подарку, Серёжа начал ими играть. Он играл ими как-то неправильно, и разломал некоторые параллелепипеды на кубики. После этого, ему не понравились кубики, на которых написана буква В, поэтому он их все потерял. Таким образом, у него остались несколько параллелепипедов, на которых написано АВ или ВА и несколько кубиков с буквой А.

После этого, пока родителей не было дома, Серёжа нашёл клей. Как и любой сообразительный мальчик, он сразу понял его предназначение и склеил все имеющиеся у него кубики и параллелепипеды в несколько линий $1 \times 1 \times l_i$. Вернувшиеся родители заинтересовались, могли ли данные конструкции получиться из имеющихся у Серёжи объектов, или нет.

Формат входных данных

В первой строке входного файла задано число N ($1 \leq N \leq 10$) — количество линий, в которые склеил кубики Серёжа.

Далее, в каждой из следующих N строк входного файла задана строка, состоящая только из букв А и В — полученная Серёжей линия из кубиков.

Суммарная длина всех строк во входном файле не превышает 100 000 символов.

Формат выходных данных

В выходной файл выведите n строк. В i -й строке выходного файла должно быть написано «YES», если Серёжа мог получить i -ю линию, и «NO» в противном случае.

Примеры

abbaa.in	abbaa.out
5	YES
A	NO
B	YES
ABBA	NO
BABBA	YES
ABBAА	

Задача Е. Простая задача

Имя входного файла: `easy.in`
Имя выходного файла: `easy.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вам дано число N . Посчитайте количество способов изменить в нём ровно две цифры на различных позициях (при этом не разрешается менять цифру на неё же саму) так, чтобы получившееся число делилось на 42. Способы, в которых в числе появляются ведущие нули, разрешены.

Формат входных данных

В первой строке входного файла дано число N ($1 \leq N \leq 10^{100000}$)

Формат выходных данных

В единственную строку выходного файла выведите ответ на задачу.

Примеры

<code>easy.in</code>	<code>easy.out</code>
43	2
100	7

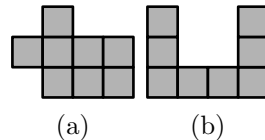
Замечание

В первом примере можно получить числа 00, 84

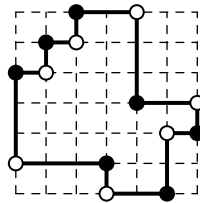
Задача F. Выпуклые пермутомино

Имя входного файла: `permutominoes.in`
Имя выходного файла: `permutominoes.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Полимино — это связное множество клеточек на клетчатой доске. Полимино называется *выпуклым*, если каждая строка и каждый столбец полимино являются связными. Например, полимино на картинке (a) является выпуклым, а полимино, изображенное на картинке (b) не является выпуклым.



Полимино называется *пермутомино*, если оно состоит из $2 \cdot n$ вершин, все его вершины имеют координаты от 1 до n , и в нем нет ни двух вертикальных отрезков с одинаковой x -координатой, ни двух горизонтальных отрезков с одинаковой y -координатой. Картинка ниже иллюстрирует пермутомино с 14 вершинами. Заметим, что это пермутомино является выпуклым.



Вам дано число n . Ваша задача состоит в том, чтобы посчитать количество различных выпуклых пермутомино с $2 \cdot n$ вершинами. Два пермутомино считаются равными, если их можно наложить одно на другое, при этом повороты и отражения не разрешаются.

Формат входных данных

Единственное число n ($2 \leq n \leq 30$).

Формат выходных данных

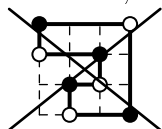
Одно число — количество выпуклых пермутомино с $2 \cdot n$ вершинами.

Примеры

<code>permutominoes.in</code>	<code>permutominoes.out</code>
2	1

Замечание

Помните, что вам следует считать только выпуклые пермутомино.



Задача G. Наибольшая общая возрастающая подпоследовательность

Имя входного файла: `lcis.in`
Имя выходного файла: `lcis.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Даны две последовательности чисел — a и b . Нужно найти наибольшую общую возрастающую подпоследовательность. Более формально: такие $1 \leq i_1 < i_2 < \dots < i_k \leq a.n$ и $1 \leq j_1 < j_2 < \dots < j_k \leq b.n$, что $\forall t : a_{i_t} = b_{j_t}, a_{i_t} < a_{i_{t+1}}$ и k максимально.

Формат входных данных

На первой строке целые числа n и m от 1 до 3000 — длины последовательностей. Вторая строка содержит n целых чисел, задающих первую последовательность. Третья строка содержит m целых чисел, задающих вторую последовательность. Все элементы последовательностей — целые неотрицательные числа, не превосходящие 10^9 .

Формат выходных данных

Выведите одно целое число — длину наибольшей общей возрастающей подпоследовательности.

Примеры

lcis.in	lcis.out
6 5 1 2 1 2 1 3 2 1 3 2 1	2