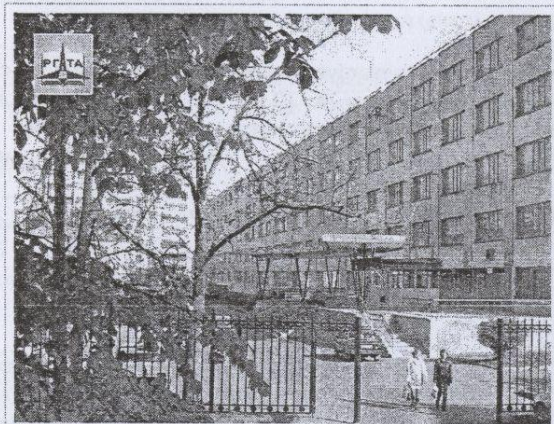




Rybinsk State Aviation Technical University

Program Committee

- Sergey G. Volchenkov,
- Vladimir N. Pinaev,
- Michael Y. Kopachev,
- Andrey Mirzoyan,
- Alexander Kiselyov,
- Dmitry Shalaev.

© RSATU, 2017 (<http://www.rsatu.ru>)



Каждый из нас лишь выиграет, создавая время от времени «игрушечные» программы с заданными искусственными ограничениями, заставляющими нас до предела напрягать свои способности. ... Искусство решения мини-задач на пределе своих возможностей оттачивает наше умение для реальных задач.

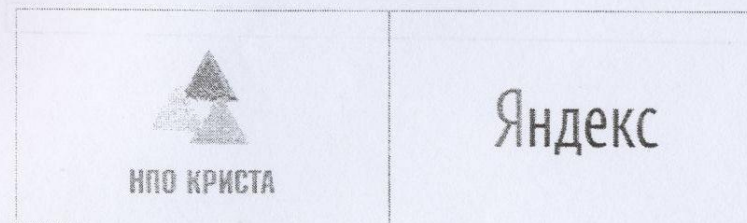
Д. Кнут

Региональная командная школьная олимпиада по программированию Центральный регион России Рыбинск, 17–19 октября 2017

A. Магический квадрат.....	2
B. Прямоугольники.....	3
C. Волшебный футбол.....	5
D. «38 попугаев».....	7
E. Черный ящик.....	9
F. Шпионаж.....	10
G. Экзамен.....	12
H. Дырокол.....	13
I. Ханойская башня.....	14

Input
Output
Memory: 256 Mb,

standard input
standard output
Time: 1 sec / test



А. Магический квадрат

Нормальный магический квадрат — это квадратная таблица $n \times n$, заполненная натуральными числами от 1 до n^2 таким образом, что:

- сумма чисел в каждой строке, каждом столбце и на обеих диагоналях одинакова;
- каждое из чисел используется ровно один раз.

Нормальные магические квадраты существуют для всех размеров кроме $n=2$. При $n=1$ квадрат тривиален, так как состоит из одного числа 1. На рисунке ниже показан нормальный магический квадрат размера 3×3 .

2	7	6
9	5	1
4	3	8

Вы можете потренироваться в построении нормальных магических квадратов в свободное время, а пока напишите программу, возвращающую сумму чисел первой строки квадрата $n \times n$.

Входные данные

Во входном файле записано одно натуральное число n — размер квадрата.

Выходные данные

В выходной файл запишите сумму элементов первой строки нормального магического квадрата размера $n \times n$.

Ограничения

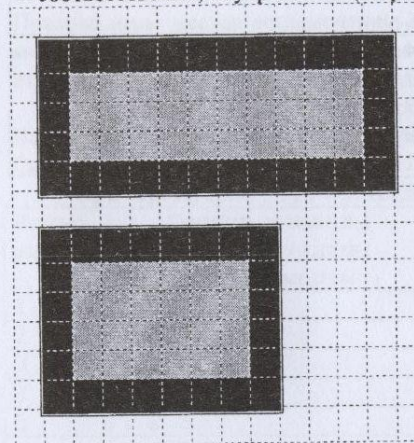
$1 \leq n \leq 1000$, $n \neq 2$, n — натуральное число.

Пример

Input	Output
3	15
1	1

В. Прямоугольники

Рассмотрим прямоугольники, нарисованные по линиям клетчатой бумаги. Клетки прямоугольника, примыкающие к его сторонам назовем внешними (на рисунке ниже обозначены черным цветом), а клетки, не касающиеся его сторон, — соответственно, внутренними (на рисунке обозначены серым цветом).



Определите, сколько существует прямоугольников, у которых внешних клеток ровно в n раз меньше, чем внутренних, а также выведите размеры всех таких прямоугольников, в порядке возрастания наименьшей стороны.

Входные данные

В строке ввода дано единственное натуральное число n — соотношение количества внешних и внутренних клеток прямоугольника.

Выходные данные

В ответе сначала напечатайте k — количество прямоугольников, удовлетворяющих условию, а далее в k последующих строках напечатайте пары чисел — стороны подходящих прямоугольников, упорядоченные по возрастанию наименьшей стороны. Размеры прямоугольника указываются в виде пары чисел, разделенных пробелом — сначала меньшая сторона, потом большая.

Ограничения

$1 \leq n \leq 10^9$

Пример

Input	Output
1	2 5 12 6 8
2	4 7 30 8 18 9 14 10 12

С. Волшебный футбол

В Волшебной стране очень популярен Волшебный футбол – игра, когда бить по мячу можно исключительно с помощью своих волшебных способностей. К сожалению, такое использование способностей сильно истощает игроков, поэтому замены в волшебном футболе – обычное дело. Правила игры жестко регламентируют, что каждый состав играет ровно одну минуту. При смене составов заменяется ровно один игрок. В течение одной игры недопустимы совпадающие составы.

В команде Изумрудного города играют n игроков. По правилам Волшебного футбола на поле в любой момент должны находиться k игроков. Сама игра длится t минут.

Пример. Пусть $n = 5$, $k = 3$, тогда возможны всего 10 составов.

Номер минуты	Состав команды (игроки заданы своими номерами)	Замена в конце минуты
1	1 2 3	меняем 3 на 4
2	1 2 4	меняем 2 на 3
3	1 3 4	меняем 4 на 5
4	1 3 5	меняем 3 на 2
5	1 2 5	меняем 2 на 4
6	1 4 5	меняем 1 на 3
7	3 4 5	меняем 3 на 2
8	2 4 5	меняем 4 на 3
9	2 3 5	меняем 5 на 4
10	2 3 4	Игра окончена

Напишите программу, которая по заданному размеру команды n , числу игроков на поле k и длительности матча t минут сформирует график замен, или сообщит, что решение не существует.

Если таких графиков окажется несколько, то решением является любой из них.

Входные данные

Входной файл содержит три целых числа n , k и t , разделенных пробелами.

Выходные данные

Первая строка выходного файла содержит число t , если решение есть или 0 – если решение не существует.

Если решение найдено, то далее следуют t строк, каждая из которых содержит описание состава команды. Строка описания состава состоит из k номеров игроков, разделенных пробелами. Номера игроков упорядочены строго по возрастанию. Составы команд не должны повторяться и в соседних строках должны отличаться ровно одним игроком.

Ограничения

$$0 < k \leq n \leq 100, 1 \leq t \leq 10^5.$$

Игроки нумеруются числами от 1 до n .

Пример

Input	Output
5 3 10	10 1 2 3 1 2 4 1 3 4 1 3 5 1 2 5 1 4 5 3 4 5 2 4 5 5 2 3 4
2 1 3	0

D. «38 попугаев»

Как известно из культового мультфильма, длина удава составляет 5 мартышек, 2 слоненка или 38 попугаев и одно попугайское крылышко, потому что звери не могут использовать метрическую систему. Добрый доктор Айболит решил проверить, можно ли подобрать подходящую единицу измерения для пациентов в очереди на прием.

Лечение занимает большую часть времени доктора, поэтому заниматься измерениями он может только в свободное время.

Приходя на прием к доктору, звери выстраиваются в очередь. Когда пациент встает в очередь, помощник доктора измеряет его длину и заносит результат в список. Прием ведется в порядке очереди. После окончания приема пациент из списка вычеркивается. Периодически Айболит смотрит в список и пытается определить, есть ли в списке такой зверь, которым можно измерить всех остальных (еще не вычеркнутых зверей списка) так, чтобы их длины оказались целыми числами.

Напишите программу, которая поможет доктору Айболиту.

Входные данные

Первая строка содержит число n — количество действий со списком пациентов. Далее следуют n строк, каждая из которых описывает одно действие со списком пациентов. Всего возможны действия трех типов:

- постановка пациента в очередь — описывается строкой, начинающейся со знака «+», за которым (без пробелов) указывается длина нового пациента;
- удаление пациента из очереди — описывается строкой состоящей из единственного знака «-»;
- поиск пациента на роль единицы измерения — описывается строкой, состоящей из единственного знака «?». Поиск производится в момент, когда очередь не пуста.

Выходные данные

Выходной файл должен содержать ровно по одной строке на каждый поиск. Если в списке есть зверь, в котором можно измерить всех остальных зверей списка, то строка должна начинаться с заглавной буквы «Y», за которой должна следовать длина этого зверя.

Если такого зверя нет, то строка должна содержать заглавную букву «N».

Ограничения

$$1 \leq n \leq 10^5.$$

Длина пациента — целое положительное число не превосходящее 10^9 .

Пример

Input	Output
10	Y2
+4	N
+2	Y3
?	Y1
+3	
?	
-	
-	
?	
+1	
?	

Е. Черный ящик

При расшифровке данных бортовых самописцев обнаружилось, что в результате ошибки в микрокоде одного из контроллеров на ленту записывались искаженные данные.

Проведенный анализ показал, что вместо записи на ленту n -значного двоичного числа $(x_1x_2x_3\dots x_n)$ записывался n -значный результат многократного суммирования исходного значения с одновременным сдвигом влево $(x_1x_2x_3\dots x_n + x_2x_3\dots x_n0 + x_3\dots x_n00 + \dots + x_n00\dots0)$.

Так как разрядность регистра для хранения промежуточных данных составляет n бит, то старшие биты, полученные при сдвигах и сложениях, отбрасывались.

Например, исходное число 0101 после такого преобразования превращается в 1011 $(0101 + 1010 + 0100 + 1000 = 1011)$.

Более того, результат суммирования был записан в порядке от младших битов к старшим (т.е. слева самый младший бит, справа – самый старший), поэтому число из примера выше будет выглядеть так: 1101.

Напишите программу, которая по закодированному числу восстановит исходное и выведет его в порядке от младших битов к старшим.

Входные данные

Первая строка входного файла содержит число n – количество битов в числе. Вторая строка содержит результат суммирования – двоичное число из n бит, записанное начиная с младшего бита.

Выходные данные

Выходной файл должен содержать единственную строку – исходное число из n бит, записанное начиная с младшего бита.

Ограничения

$$1 \leq n \leq 2^{1\,000\,000}$$

Пример

Input	Output
4	1010
1101	

Ф. Шпионаж

Хорошо известно, что большая часть разведывательной информации получается из открытых источников. Согласно этому постулату, в Сказочном королевстве была организована аналитическая служба. Первой задачей перед службой, была поставлена цель – определить маршруты транспортировки волшебных кристаллов в соседнем Волшебном царстве. В результате работы были установлены следующие факты:

- кристаллы могут перевозиться только между городами;
- рядом с одним из городов есть кристаллическая шахта – источник кристаллов;
- в одном из городов кристаллы перерабатываются – именно в нем перевозка кристаллов завершается;
- с целью затруднения перехвата, маршруты перевозки кристаллов часто меняются;
- кристаллы перевозятся только по определенным дорогам;
- для каждой дороги раз и навсегда определено единственное направление перевозки кристаллов;
- пользуясь дорогами, никакой кристалл нельзя вернуть в город, в котором он уже был, вне зависимости от того, из какого города она начинается;
- любые два города связаны не более чем одной дорогой.

Также, удалось получить информацию о том, сколько маршрутов существует между источником кристаллов и остальными городами. Вашей задачей будет восстановить возможную карту перевозок.

Всего в Волшебном царстве существует n городов, попарно связанных дорогами. Города, пронумерованы числами от 1 до n . Для перевозки волшебных кристаллов используются только некоторые из дорог. Установлено, что кристаллическая шахта расположена в городе с номером m . Количество возможных маршрутов из города m в остальные города представлено в виде вектора D , где элемент $D[i]$ хранит количество различных маршрутов от города m до города i .

Напишите программу, которая по заданным числам n , m и вектору D построит возможную карту перевозок. Если таких карт несколько, то построить можно любую из них.

Входные данные

Первая строка входного файла содержит целые числа n и m , разделенные пробелом. Далее следует строка, содержащая числа $D[1]$, $D[2]$, ... $D[n]$, разделенные пробелами.

Выходные данные

Первая строка выходного файла содержит единственное целое число k – количество дорог, задействованных при перевозке кристаллов. Далее следуют k строк, каждая из которых содержит пару чисел – номера городов A_i , B_i , которая означает, что кристаллы могут перевозиться из города A_i в город B_i .

Ограничения

$$2 \leq n \leq 60, 1 \leq m \leq n, 0 \leq D[i] \leq 2^{62}.$$

Пример

Input	Output
5 1	6
0 1 1 3 3	1 2
	1 3
	1 4
	2 4
	3 4
	4 5

Г. Экзамен

При сдаче ЕГЭ по информатике Василий решал следующую задачу.

У исполнителя *A17* есть три команды, которым присвоены номера:

1. Прибавить к текущему значению 1
2. Прибавить к текущему значению целое x ($x > 1$)
3. Умножить текущее значение на 7

Первая команда увеличивает число на экране на 1, вторая увеличивает его на x , третья – умножает на 7.

Программа для исполнителя *A17* – это последовательность команд с номерами из диапазона 1..3. Исполнитель *A17* преобразует число на экране по заданной последовательности команд. В качестве первого значения берется всегда число 1. Обозначим через N результат выполнения программы.

Погрузившись в тему, Василий рассчитал число K – количество вариантов программы для исполнителя *A17* таких, что в результате получается некоторое число N .

Например, при $x = 5$, $N = 7$ существует $K=4$ таких программы (указаны номера команд):

- 1, 1, 1, 1, 1, 1;
- 2, 1;
- 1, 2;
- 3.

А после экзамена, Василий задумался над задачей: как по входным данным: N и K определить число x ?

Напишите программу, которая по заданным числам N и K найдет наименьшее число x , удовлетворяющее условиям задачи.

Входные данные

Первая строка входного файла содержит два целых числа N и K , разделенные пробелом.

Выходные данные

Выходной файл должен содержать единственное целое число x – наименьшее из удовлетворяющих условию. Либо 0, если такого x не существует.

Ограничения

$0 < K \leq 2^{60}$; $0 < N \leq 10^4$; $1 < x < N$.

Пример

Input	Output
7 4	5
14 3	0

Н. Дырокол

На узкой бумажной ленте, вдоль центральной оси, с шагом 1 см, необходимо пробить n одинаковых отверстий, причем n – всегда четное. Для пробивания используется дырокол, который за один раз пробивает всегда по два отверстия на расстоянии k см друг от друга. Пробить только одно отверстие таким дыроколом невозможно (нельзя ставить дырокол на уже пробитое отверстие или мимо ленты).

В наличии имеются дыроколы со всеми возможными различными расстояниями между пробиваемыми отверстиями, т.е., есть дыроколы с расстояниями $k = 1, k=2, k = 3$ и так далее. Вашей задачей будет написать программу, которая по числу n перечислит дыроколы, расстояния k которых подойдут для выполнения работы.

Входные данные

Входной файл содержит единственное целое число n .

Выходные данные

Первая строка выходного файла содержит целое число k – количество подходящих дыроколов. Вторая строка содержит k целых чисел, разделенных пробелами, и упорядоченными по возрастанию.

Ограничения

$2 \leq n \leq 10^9$,

n – четное,

$1 \leq k \leq n$.

Пример

Input	Output
8	3 1 2 4

I. Ханойская башня

Участники чемпионата по программированию, оттачивая свое мастерство, регулярно решают задачу «Ханойская башня». Все команды используют стандартный классический алгоритм перекладывания, рассчитанный на $(2^N - 1)$ ходов.

Ниже приведена соответствующая процедура на языке Паскаль.

```

Procedure Hanoi (X, Y, Z: char; N: integer);
Begin
  If N > 0 then
    Begin
      Hanoi (X, Z, Y, N-1);
      Writeln('Disk ', N, ' from ', X, ' to ', Y);
      Hanoi (Z, Y, X, N-1);
    End
  End;

```

И вот нам стали известны промежуточные результаты, которых добились команды. Все результаты представлены в виде строк длины N (число дисков), где i -ый элемент строки обозначает стержень, на котором находится i -ый диск.

Например, пусть $N=4$. Тогда запись «BVCA» означает, что первый и второй диски находятся на стержне В, третий диск – на стержне С и четвертый – на стержне А.

Для примера в таблице ниже приведены все 7 ходов для решения Ханойской башни из трех дисков.

Номер хода	Распределение дисков по стержням
0	AAA
1	BAA
2	BCA
3	CCA
4	CCB
5	ACB
6	ABV
7	BBV

Напишите программу, определяющую номер команды, выполнившей максимальное число ходов Ханойской башни. Если таких команд окажется несколько, то определите номер первой из них.

Входные данные:

Первая строка содержит целое число N – количество дисков и целое число M – количество проверяемых решений.

Последующие M строк содержат по N символов – решения соответствующих команд. Строка под номером i – это результат работы i -ой

команды. Каждая строка есть результат распределения дисков по стержням А, В и С.

Выходные данные

Одно целое число – номер команды, выполнившей максимальное число ходов. Если таких команд окажется несколько, то выводится минимальный номер.

Ограничения

Каждая строка соответствует распределению дисков по стержням после выполнения хода с номером K ($0 \leq K \leq 2^N - 1$). Все команды используют стандартный классический алгоритм перекладывания, рассчитанный на $(2^N - 1)$ ходов. Количество дисков N удовлетворяет условию $1 \leq N \leq 10^3$. Количество команд: $1 \leq M \leq 10^3$.

Input	Output
4 7 CAAA AAAA CCCB CBAA BVAA BVCA CCAA	3
3 4 AAA BBB BAA BBV	2