Задача A. Праздничная олимпиада

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

У мальчика Вити скоро день рождения, который он хочет провести с друзьями. Какой же праздник без праздничной олимпиады? Для своих друзей Витя подготовил олимпиаду по программированию.

Когда все гости собрались, именинник распределил их по m командам. Витя очень старался сбалансировать силы команд, и у него это получилось. Каждую из предложенных n задач олимпиады любая команда решает ровно за a_i минут и всегда с первой попытки. Ребята не любят распыляться, поэтому если команда берется писать задачу i, то все ее участники непрерывно решают задачу в течении a_i минут. При этом команда может начинать решать следующую задачу сразу после того, как решила предыдущую.

Однако, Витя не хотел поссорить своих друзей, поэтому исключил элемент борьбы между командами — команды не соперничают, а помогают друг другу. Целью команд на олимпиаде является решение всех задач. Таким образом каждую задачу решает только одна команда. При этом друзьям засчитывается штрафное время, равное числу минут, прошедших от начала олимпиады до момента сдачи задачи. Единственная цель, которую преследуют друзья — сдать все задачи с наименьшим суммарным штрафным временем.

Например, пусть участвуют две команды, которым предложено в олимпиаде три задачи с временами решения 5, 10 и 15 минут. Пусть первая команда решает сначала вторую, а потом первую задачу. Таким образом за решение второй задачи друзья получат 10 штрафных минут, а за решение первой 15. Вторая команда с начала олимпиады решает только третью задачу, за которую друзья получат 15 штрафных минут. В сумме друзья Вити получат 40 штрафных минут.

В то время, когда друзья будут решать задачи, Витя будет управлять порядком, в котором команды будут их решать. Помогите Вите — напишите программу, которая вычислит наименьшее суммарное штрафное время, требуемое для сдачи всех задач.

Формат входного файла

В первой строке входного файла содержатся два числа n и m — число задач и команд, соответственно ($0 \le n \le 50000$, $1 \le m \le 10000$). Во второй строке содержится n целых чисел a_i — количество времени, необходимое для решения задачи i любой из команд ($0 \le a_i \le 30$).

Формат выходного файла

Выведите в выходной файл наименьшее суммарное штрафное время, которое могут получить друзья Вити за решение всех задач.

Примеры

contest.in	contest.out
3 2 10 15 5	35
2 3 23 30	53

Задача B. Абрикосовый урожай

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

Вовсе не секрет, что абрикосы являются национальным фруктом Аброляндии. Ни один гражданин Аброляндии не может представить себе жизнь без абрикосов.

К сожалению, прошлый год оказался крайне неурожайным, что привело к печальным последствиям. Король Аброляндии Брико I понял, что больше такого нельзя допускать, и приказал вывести новые сорта абрикосов, которые должны привести к небывалой доселе урожайности.

Приказ выполнили оперативно. Для проверки урожайности посадили на поле n рядов по m абрикосовых деревьев. Агрономы после сбора урожая с экспериментального поля утверждают, что новый сорт дает невиданные результаты урожайности.

Урожайность определяется следующим образом: для каждого квадрата на поле количество собранных абрикосов должно быть не меньше некоторого количества S, заданного самим королем. Заметим, что квадрат может состоять из одного дерева.

Великий король Брико I усомнился в результатах, выданных агрономами и приказал посчитать количество квадратов, где не выполняется критерий урожайности. Но, к сожалению, аброляндские математики все время сбиваются со счета, поэтому Брико I попросил Вас написать программу, решающую его проблему.

Формат входного файла

Первая строка содержит три числа n, m, S — размеры карты экспериментального абрикосового поля и заданное королем число, соответственно ($1 \le n, m \le 30$; $0 \le S \le 10^9$). Далее идут n строк по m чисел a_{ij} - количество абрикосов на каждом дереве($0 \le a_{ij} \le 10^7$).

Формат выходного файла

В выходной файл выведите единственное число - количество квадратов, где не выполняется критерий урожайности.

crop.in	crop.out
2 3 4 1 2 3 4 5 6	3
2 3 11 1 2 3 4 5 6	6

Задача C. Лепреконское золото

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

Коренной житель Ирландии лепрекон Патрик однажды крупно поссорился со своей женой Клариссой и решил в срочном порядке убежать на остров Бора-Бора. Для этого у мудрого Патрика есть n спрятанных на одной прямой горшочков с лепреконским золотом. Ссора произошла спонтанно, поэтому Патрик не смог запастись достаточным количеством магических амулетов и талисманов, а это значит, что его магических сил хватит лишь на одну телепортацию, но зато в любое место, например — к любому из горшочков с золотом. Эту телепортацию необходимо использовать, до начала сбора горшочков с золотом.

Так как лепреконы по природе своей не очень хорошие бегуны, без помощи телепортаций Патрик может перемещаться со скоростью 1 метр в минуту. Но на один из горшочков Кларисса наложила заклинание исчезновения, и он пропадет через t минут. Помогите Патрику за минимальное время собрать все горшочки с золотом! Если он не заберет хотя бы один из них, ему не хватит золота на путешествие.

Формат входного файла

В первой строке число n — количество горшочков с золотом — и число t — время исчезновения (в минутах) одного из них ($2 \le n, t \le 100$). В следующей строке n чисел — координаты горшочков в метрах. Все числа различны и по абсолютной величине не превосходят 100. Координаты горшочков даны в порядке возрастания. В следующей строке записан номер горшочка, который исчезнет через t минут.

Формат выходного файла

В первой строке выходного файла выведите минимальное время, которое потребуется Патрику для сбора всего золота. В следующей строке выведите n чисел — порядок, в котором следует собирать горшочки.

Примеры

gold.in	gold.out
5 5 1 4 9 16 25 2	24 1 2 3 4 5
6 4 1 2 3 6 8 25 5	31 5 4 3 2 1 6

Задача D. Награды

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

На одном Очень Важном Предприятии решили наградить некоторых k его работников. Конечно же, решили сделать это в соответствии со следующей Очень Важной Процедурой.

Всех n работников выстроили в один ряд. Причем, получилось так, что каждый работник видит только своих непосредственных соседей в этом ряду. Для повышения уровня производства на Очень Важном Предприятии начальство решило сделать так, чтобы каждый награжденный считал, что наградили именно его и только его. Для этого необходимо, чтобы в ряду не было двух рядом стоящих награжденных работников.

Вам необходимо написать программу, которая будет считать количество способов раздать таким образом k наград среди n стоящих в ряд работников.

Формат входного файла

В первой и единственной строке входного файла заданы два целых числа n и k — количество работников на Очень Важном Предприятии и количество наград ($1 \le k \le n \le 20$).

Формат выходного файла

В выходной файл выведите единственное целое число — ответ на задачу.

Примеры

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

Задача E. Блокировка телефона

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

В компании по производству мобильных телефонов *SimSung* готовится выпуск смартфона нового поколения под кодовым названием *Mokia*. Маркетологи уверяют, что данный телефон прочно закрепится на рынке смартфонов и быстро вытеснит всех своих конкурентов.

Предполагается, что платформой смартфона станет всем известная операционная система *Doors Mobile 8* компании *MacroHard*. Но вот незадача: выпуск первой серии смартфонов запланирован на понедельник, а операционная система *Doors Mobile 8* находится в режиме бета-тестирования. Программисты столкнулись с проблемой при тестировании метода разблокирования телефона. Дело в том, что для разблокирования телефона используется инновационная технология *ThinkAnd-Touch*. Суть ее заключается в следующем: пользователь вводит комбинацию из трех цифр. Если соответствующие клавиши расположены в одном ряду по горизонтали или вертикали подряд, а цифры на них попарно различны, то телефон разблокируется.

Клавиатура *Mokia* выглядит следующим образом:

1	2	3
4	5	6
7	8	9
	0	

Вас нанял главный начальник *SimSung* и попросил написать программу, проверяющую, может ли данная комбинация клавиш разблокировать телефон. Не подведите его!

Формат входного файла

В единственной строке даны три цифры — комбинация клавиш, нажатая пользователем.

Формат выходного файла

Выведите «Unlocked», если телефон можно разблокировать данной комбинацией клавиш. В противном случае выведите «Locked».

Примеры

lock.in	lock.out
125	Locked
208	Locked
123	Unlocked

Задача F. Ответный матч

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

Две популярные футбольные команды А и Б решили выявить сильнейшего между собой. Для этого они решили провести двухматчевое противостояние — сначала сыграть матч на поле А, затем на поле Б.

Противостояние выигрывает та команда, которая в сумме забьет больше голов, чем соперник. Если же команды забили одинаковое число мячей в ворота соперника, то победителем признается команда, забившая большее число голов в гостях (на стадионе соперника). Если же и эти величины совпадают, то команды исполняют серию пенальти.

Второй матч противостояния на поле Б на радио «Слушаем внимательно» комментировал ведущий Вася. Во время трансляции его слушатели регулярно интересовались не только текущим результатом матча, который Вася зачитывал с табло, но и возможностью такого исхода встречи, что будет исполнена серия пенальти (как самое захватывающее футбольное зрелище).

Математические способности Васи оставляли желать лучшего, и он попросил Вас написать программу, которая, зная результат первого матча и текущий счет на табло во время второго матча, поможет определить, смогут ли зрители наблюдать серию пенальти.

Формат входного файла

Первая строка входного файла содержит результат матча А — Б (на поле команды А) в формате «a:b», где a — число мячей, забитых хозяевами поля, а b, соответственно, гостями ($0 \le a, b < 10$). Вторая строка в том же формате содержит текущий счет на табло в матче Б — А.

Формат выходного файла

В выходной файл выведите слово «`YES`», если команды могут закончить встречу со счетом, после которого результат противостояния определиться в серии пенальти, иначе — выведите слово «`NO`».

Примеры

match.in	match.out
2:1 0:0	YES
2:1 0:2	NO

Задача G. Вещественное число

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

Вася продолжает писать свою операционную систему. Теперь ему необходимо написать модуль поддержки вывода вещественных чисел.

Данный модуль должен, принимая на вход вещественное число, лежащее в пределах от нуля до единицы, и необходимое число цифр после десятичной точки, выводить это число с заданным числом цифр после десятичной точки при необходимости, округляя его, либо дописывая в конец необходимое число нулей.

Округление при этом происходит следующим образом: если следующая за округляемой цифра меньше 5, то округляемая цифра не меняется, если же следующая цифра больше или равна 5, то округляемая цифра увеличивается на единицу. При этом, если она была равна 9, то она превращается в 0, а предыдущая перед ней увеличивается на единицу и так далее.

Вам необходимо написать программу, реализующую данный модуль.

Формат входного файла

В первой строке входного файла задано вещественное число x ($0 \le x < 1$). Число содержит не более 100 цифр после десятичной точки и обязательно начинается с нуля и, следующей после него, десятичной точки (0.).

Вторая строка входного файла содержит целое число k ($1 \le k \le 100$) — необходимое число цифр после десятичной точки.

Формат выходного файла

В выходной файл выведите вещественное число x с k цифрами после десятичной точки в аналогичном формате.

Примеры

significant.in	significant.out
0.31415 3	0.314
0.99999 3	1.000
0.566 5	0.56600

Задача H. Треугольник

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

Задан треугольник и число S. Необходимо на сторонах треугольника выбрать три точки так, чтобы образуемый ими треугольник имел площадь S.

Формат входного файла

В первой строке входного дано шесть целых чисел x_1, y_1, x_2, y_2, x_3, y_3 — координаты вершины треугольника, все координаты по модулю не превосходят 1000. Во второй строке находится целое число S — площадь, которую необходимо получить ($0 \le S \le 10^9$).

Формат выходного файла

Если точки выбрать нельзя, то выведите «No solution», иначе выведите шесть чисел x'_1, y'_1, x'_2, y'_2, x'_3, y'_3 —координаты вершин нового треугольника. Точки должны отстоять от сторон треугольника не более чем на 10^{-5}. Площадь выведенного треугольника должна отличатся от искомой не более чем на 10^{-5}.

Примеры

triangle.in	triangle.out
0 0 7 0 0 7 6	3.0 0.0 3.0 4.0 0.0 4.0
0 0 3 0 0 4 7	No solution

В первом примере ответ может выглядеть так: