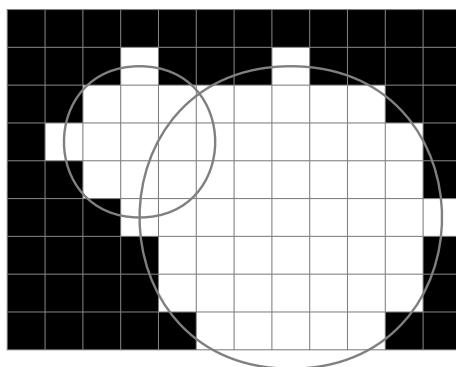


## Задача А. Круги на экране

Имя входного файла: `circles.in`  
Имя выходного файла: `circles.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 Мб

Вчера Петя написал программу, которая рисует круги на чёрно-белом экране с разрешением  $w \times h$  пикселей. Изначально весь экран был чёрным, но потом Петя запустил свою программу, и она нарисовала на нём  $n$  белых кругов.

У каждого пиксела есть координаты: левый верхний угол имеет координаты  $(0, 0)$ , а правый нижний — координаты  $(w - 1, h - 1)$ . Круг с центром в  $(x_c, y_c)$  радиуса  $r$  содержит те пиксели  $(x, y)$ , для которых выполнено неравенство  $\sqrt{(x_c - x)^2 + (y_c - y)^2} \leq r$ . Если круг вылезает за пределы экрана, то части круга, оказавшиеся вне экрана, нигде не отображаются. Если какой-то пиксел лежит в двух или более кругах, он всё равно будет покрашен в белый цвет.



Картинка получилась красивая, и Петя решил скопировать её себе на стену. Изначально обои на стене белые; Петя может покрасить какие-то части стены в чёрный цвет. Осталось только узнать, сколько краски для этого понадобится. Копирование будет производиться попиксельно, поэтому необходимо узнать, сколько чёрных пикселей осталось на экране. Напишите программу, которая вычисляет это количество.

### Формат входных данных

В первой строке входного файла записаны три числа:  $w$ ,  $h$  и  $n$  ( $1 \leq w, h \leq 200$ ;  $1 \leq n \leq 100$ ). Каждая из следующих  $n$  строк задаёт круг в формате  $x_i y_i r_i$  ( $0 \leq x_i < w$ ;  $0 \leq y_i < h$ ;  $0 \leq r_i \leq 40\,000$ ); здесь  $(x_i, y_i)$  — это координаты центра  $i$ -го круга, а  $r_i$  — его радиус.

### Формат выходных данных

Выведите в выходной файл одно число — количество оставшихся на экране чёрных пикселей.

### Примеры

<code>circles.in</code>	<code>circles.out</code>
5 3 2 1 1 1 3 1 1	6
12 9 2 3 3 2 7 5 4	51

### Замечание

Картинка выше соответствует второму примеру.

## Задача В. Чемпионат мира

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

На бесконечном поле стоит экран и проектор, через который транслируют на экран в прямом эфире чемпионат мира по программированию. Петя, который находится в некоторой точке данного поля, хочет побыстрее посмотреть трансляцию.

Будем считать, что экран является отрезком  $AB$  с концами  $(0, 0)$  и  $(w, 0)$ , изображение проецируется на него из точки с координатами  $(\frac{w}{2}, 5000)$ , а Петя стоит в точке  $(x, y)$ . Из точки  $C$  можно наблюдать за игрой в том и только в том случае, если у точки  $C$  положительная ордината, а  $\angle ABC$  и  $\angle BAC$  не превосходят  $135^\circ$ .

Петя может перемещаться по полю как угодно, но не может проходить сквозь экран. Какое наименьшее расстояние понадобится пройти Петя, чтобы попасть в точку, откуда он сможет наблюдать за чемпионатом?

### Формат входных данных

В первой строке через пробел записаны целые числа  $w, x, y$  ( $1 \leq w \leq 10^4; |x|, |y| \leq 10^4$ ) — ширина экрана и координаты Петинного начального положения.

### Формат выходных данных

Выведите наименьшее расстояние, которое Пете понадобится пройти, чтобы посмотреть чемпионат, с точностью не менее четырех знаков после десятичной точки.

### Примеры

worldcup.in	worldcup.out
5 1 4	0
5 9 -1	3.5355
12 4 -3	5.00000



## Задача С. Старая крепость

Имя входного файла: `castle.in`  
Имя выходного файла: `castle.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 Мб

В одной далекой стране ученые обнаружили странное скопление камней. Изучив его, ученые пришли к выводу, что это части старой крепостной стены, имевшей форму окружности. К сожалению, время и вандалы разрушили некоторые части стены.

Чтобы защитить оставшиеся фрагменты стены и продолжить их изучение в спокойной обстановке, ученые хотят обнести фрагменты стены забором из колючей проволоки. Если сделать отдельный забор для каждого фрагмента, будет неудобно переходить от одного фрагмента к другому, поэтому ученые хотят сделать один общий забор, окружающий все фрагменты.

Помогите ученым посчитать минимальную возможную длину забора, чтобы они знали, сколько просить колючей проволоки.

### Формат входных данных

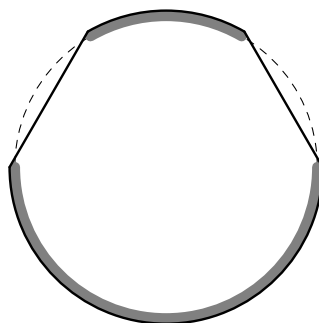
Во входном файле задано два натуральных числа: число фрагментов  $n$  ( $1 \leq n \leq 180$ ) и радиус крепости  $r$  ( $1 \leq r \leq 100$ ). Далее следует  $n$  пар целых чисел, описывающих сохранившиеся фрагменты стены:  $a_i, b_i$  — углы в градусах, соответствующие началу и концу фрагмента. Углы отмеряются от направления на север из центра крепости, против часовой стрелки ( $0 \leq a_i, b_i < 360, a_i \neq b_i$ ). Каждый фрагмент от начального угла к конечному также проходит против часовой стрелки. Фрагменты не имеют общих точек.

### Формат выходных данных

Выведите минимальную возможную длину забора. Ответ должен отличаться от правильного не более, чем на  $10^{-3}$ .

### Пример

<code>castle.in</code>	<code>castle.out</code>
2 10 330 30 90 270	61.8879020479



## Задача D. Математики и скобки

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

Встретились однажды три математика ...

- Первый математик написал мелом на доске скобочную последовательность.
- Второму математику стало интересно, существует ли циклический сдвиг, превращающий эту последовательность в правильную.
- Третий же математик, немного подумав, сказал, сколько таких сдвигов существует.

Вам известна скобочная последовательность, записанная первым математиком. Найдите число, которое произнёс третий математик.

Циклическим сдвигом строки называется перенос некоторого (возможно, нулевого) количества символов из конца строки в её начало без изменения их порядка.

### Формат входных данных

В единственной строке дана скобочная последовательность, записанная первым математиком. Длина последовательности не равна нулю и не превышает 100000 символов.

### Формат выходных данных

Выведите количество циклических сдвигов, превращающих записанную скобочную последовательность в правильную.

### Примеры

<code>brackets.in</code>	<code>brackets.out</code>
<code>) ( (</code>	1
<code>) ( ( (</code>	2
<code>(</code>	1

## Задача Е. Ралли

Имя входного файла: rally.in  
Имя выходного файла: rally.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 Мб

Вася участвует в автомобильном ралли-рейде одним из этапов которого является ночное ориентирование по лесу. Само по себе ориентирование сложности для Васи не представляет, тем более что двигаться разрешается только по дорогам, которые соединяют лесные поляны. Однако по правилам соревнований на каждой дороге задана минимальная и максимальная скорость с которой по ней разрешается двигаться. Цель соревнования состоит в том, чтобы добраться со стартовой поляны до финишной за заранее известное время  $T$ . Если же время прохождения дистанции участником отличается от  $T$ , то ему начисляется штраф, равный отклонению его времени от времени  $T$  в секундах. Поскольку дороги достаточно узкие, а участников много, то по каждой дороге разрешено движение только в одном направлении и трасса организована таким образом, чтобы участник не мог проехать по одной и той же дороге более одного раза. Вася хочет выбрать такой маршрут движения и скорости на каждой из входящих в него дорог, чтобы проехать трассу ровно за время  $T$ , а если это невозможно, то минимизировать полученный штраф. Помогите ему в этом.

### Формат входных данных

В первой строке входного файла даны 5 целых чисел: общее количество полян  $N$ , общее количество дорог  $M$ , требуемое время прохождения маршрута  $T$ , номер стартовой поляны  $S$  и номер поляны  $F$ , на которой расположен финиш ( $1 \leq N \leq 5000$ ,  $1 \leq M \leq 30000$ ,  $0 \leq T \leq 10000$ ,  $1 \leq S, F \leq N$ ). В следующих  $M$  строках даны описания дорог по одной на строке. Каждая дорога задается четырьмя целыми неотрицательными числами в следующем порядке: номер поляны  $s_i$ , на которой дорога начинается, номер поляны  $f_i$ , на которой дорога заканчивается, минимальное и максимальное время  $tmin_i$  и  $tmax_i$  в секундах, за которое дорогу разрешается проехать (зная длину дороги, минимальную и максимальную разрешенную скорость, Вася уже сам вычислил время, которое он может потратить на преодоление дороги,  $1 \leq s_i, f_i \leq N$ ,  $0 \leq tmin_i \leq tmax_i \leq 100000$ ).

### Формат выходных данных

В случае, если Васе удастся проехать трассу за время  $T$ , в единственной строке выходного файла выведите YES. В противном случае в первой строке выведите NO, а во второй строке время за которое Вася должен проехать трассу для того, чтобы минимизировать штраф, а в случае, если трассу проехать невозможно, выведите -1. Если существует несколько вариантов, при которых Вася получает одинаковый штраф, выведите любой из них.

### Пример

rally.in	rally.out
5 5 11 1 5 1 2 1 2 2 4 7 8 1 3 3 4 3 4 5 6 4 5 1 2	YES
3 3 3 2 1 3 1 1 2 2 3 3 4 2 1 1 2	NO 4
4 4 5 4 1 1 2 1 2 2 3 2 3 3 1 3 4 3 4 5 6	NO -1