

## Задача А. Интересная игра

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

Вася играет в очень интересную игру «Jump». На специальной дорожке в ряд расположено несколько батутов. Батуты бывают двух типов: те, которые рассчитаны на прыжки в высоту, и те, которые рассчитаны на прыжки в длину. Игрок прыгает по батутам слева направо. При этом, первый свой прыжок он обязан сделать на любом батуте, который рассчитан на прыжок в длину (чтобы хорошенько разогнаться), после этого подпрыгнуть на батуте, который предназначен для прыжков в высоту (теперь он с разгона сможет подпрыгнуть очень высоко!). Игра ограничена по времени, поэтому игрок прыгает только на два батута.

Ваша задача состоит в том, чтобы определить, сколько существует различных способов сыграть в игру. Два способа считаются различными, если они различаются хотя бы одним батутом (для прыжков в длину или для прыжков в высоту).

### Формат входного файла

В единственной строке входных данных содержится описание дорожки. Описание состоит из нескольких символов 'a' и 'b': 'a' обозначает батут для прыжка в длину, а 'b' — в высоту. Батуты перечислены слева направо вдоль направления дорожки. Общее число батутов не превосходит 75 000. На дорожке есть хотя бы один батут.

### Формат выходного файла

Выведите одно число — искомое число способов пройти игру.

### Примеры

<code>jump.in</code>	<code>jump.out</code>
abab	3

### Note

В примере из условия есть три варианта пройти игру:

- первый прыжок в длину на первом батуте, второй прыжок — на втором батуте;
- первый прыжок в длину на первом батуте, второй прыжок — на четвёртом батуте;
- первый прыжок в длину на третьем батуте, второй прыжок — на четвёртом батуте.

## Задача В. Сортировка

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

Учительница по программированию задала Вовочке задачу — отсортировать массив из  $N$  различных чисел по возрастанию.

Вовочка поступает так: он просматривает массив чисел слева направо, и, если замечает два элемента, стоящих рядом, таких, что правый меньше левого, он меняет их местами. Так он поступает, пока массив не будет отсортирован.

Но Вовочка — очень ленивый ученик. В какой-то момент ему надоело сортировать числа, и он решил посчитать, сколько ещё описанных выше обменов нужно сделать. Помогите ему.

### Формат входного файла

В первой строке входных данных находится натуральное число  $N$  ( $1 \leq N \leq 1500$ ). Во второй строке через пробел вводится  $N$  различных целых чисел, каждое из которых не меньше 1 и не больше 10000.

### Формат выходного файла

Выведите одно число — искомое количество обменов.

### Примеры

<code>inv.in</code>	<code>inv.out</code>
5 1 2 3 5 4	1
3 3 2 1	3

## Задача С. RPQ (Range Permutation Query)

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

От некоторых школ в командной олимпиаде по информатике участвует очень много команд. Учитель одной из таких школ раздал для регистрации своим командам номера: 1, 2, 3 и т. д. Для того чтобы проверить, все ли команды зарегистрировались, учитель выписал из таблицы регистрации только номера команд своей школы, но в том порядке, в котором команды регистрировались.

После нелёгких подсчётов оказалось, что зарегистрировались все. Но в процессе решения этой задачи учитель сформулировал следующую: сколькими способами можно выбрать стоящие подряд в этом списке  $K$  номеров команд, чтобы они образовывали некоторую перестановку чисел от 1 до  $K$ ? Например, если от школы участвуют всего три команды, то при порядке регистрации 3 1 2 таких способов будет три (для  $K = 1, 2, 3$ ), а при регистрации в порядке 2 3 1 — всего два (для  $K = 1$  и  $K = 3$ ).

### Формат входного файла

В первой строке входных данных находится одно число  $N$  ( $1 \leq N \leq 200$ ) — количество команд, участвующих в олимпиаде от этой школы. Во второй строке находятся  $N$  натуральных чисел от 1 до  $N$  в том порядке, в котором команды регистрировались на олимпиаду.

Гарантируется, что каждое число встречается в этой строке ровно один раз.

### Формат выходного файла

Выведите одно число, обозначающее число способов выбрать из списка несколько стоящих подряд команд, удовлетворяющих условию задачи.

### Примеры

<code>rpq.in</code>	<code>rpq.out</code>
3 2 3 1	2

## Задача D. Календарь

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

Джонни работает в министерстве финансов одной небольшой страны. На этот раз он решил очень тщательно распланировать бюджет. Для этого ему первым делом необходимо выяснить, сколько же будет выходных дней в каждом интересующем его году (без учёта праздников, которые и в этой стране то и дело переносят).

Так как Джонни смотрит в будущее, то его интересуют лишь года, которые ещё не наступили. Но он верит в успехи биоинформатики, касающиеся увеличения продолжительности жизни, и хочет рассчитать всё заранее, поэтому его интересуют и очень отдалённые даты. При этом, Джонни уже вычислил для интересующего его года, какой день недели приходится на первое января этого года.

### Формат входного файла

В единственной строке входных данных содержатся два целых числа: год  $Y$ , который интересует Джонни, ( $2013 \leq Y \leq 2100$ ) и номер дня недели  $D$ , на который приходится первое января этого года ( $1 \leq D \leq 7$ ).  $D = 1$  означает понедельник,  $D = 2$  — вторник и т. д.

### Формат выходного файла

В единственной строке выведите количество выходных дней в соответствующем году.

### Примеры

<code>days.in</code>	<code>days.out</code>
2013 2	104

### Note

Напомним, что в неделе семь дней, выходными считаются суббота и воскресенье. В невисокосных годах 365 дней, в високосных — 366. Год называется високосным, если он делится на 400, или если он делится на 4, но не делится на 100.

## Задача Е. Поливальная машина

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

Город  $N$ , в отличие от города  $M$ , расположен на склоне одного холма. Чистоту улиц этого города обеспечивает единственная поливальная машина. Бензин нынче дорог, поэтому движение муниципального транспорта «в гору» признано слишком расточительным.

Карта города представляет собой прямоугольник, разбитый на  $H \times W$  клеток, где  $H$  и  $W$  — высота и ширина карты в клетках соответственно. Часть клеток заняты зданиями, остальные соответствуют улицам и площадям, которые и требуется помыть.

Поливальная машина начинает свой путь в одной из клеток самого верхнего ряда, не занятой зданиями. Она может полить асфальт в текущей клетке и переместиться в любую из двух соседних клеток этого же ряда, не занятых зданиями. Объехать здания, не поднимаясь при этом в гору, невозможно. Поливальная машина также может переместиться в соседнюю по вертикали свободную клетку из нижнего ряда.

Помогите узнать экономному муниципалитету, какое максимальное количество свободных клеток поливальная машина сможет помыть, не поднимаясь при этом в гору?

Учтите, что структура города такова, что в каждом ряду свободные клетки образуют один непрерывный отрезок.

### Формат входного файла

В первой строке входного файла находятся натуральные числа  $H$  и  $W$  — высота и ширина карты города ( $1 \leq W \leq 300$ ,  $1 \leq H \leq 300$ ).

Каждая из следующих  $H$  строк содержит  $W$  символов ‘#’ и ‘.’, означающих, соответственно, клетки со зданиями и без.

### Формат выходного файла

Выведите единственное натуральное число — максимальную площадь, которую может обработать поливалка.

### Примеры

<code>water.in</code>	<code>water.out</code>
8 8 ###...## #...#### ##.##### ##.##### .....## ####...# #...#### #.....##	18

## Задача F. Фотооптимизация

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

На Международной олимпиаде по информатике некоторые участники, конечно же, получают удовольствие именно от решения предложенных задач, но большинство — от полученных в новой стране впечатлений. Впечатления принято запечатлевать на фотоаппарат. Участник Т решил подойти к процессу съёмок с научной (по его мнению) точки зрения. Он желает заснять сразу два интересных объекта, местоположение каждого из которых на земле мы будем описывать с помощью отрезка. Т выбирает точку для съёмок так, чтобы площадь двух треугольников, образованных концами соответствующих отрезков и выбранной точкой была одинаковой. Треугольники при этом должны быть невырожденными.

Помогите Т с выбором такой точки. Возможность заснять сразу два объекта при этом анализировать не нужно, мы лишь действуем в рамках модели, сформулированной Т. Задача упрощается тем, что каждый из отрезков, описывающих объекты, оказался параллельным одной из осей координат (возможно, каждый своей).

### Формат входного файла

В первой строке входного файла находятся 4 целых числа  $x_1, y_1, x_2, y_2$ , характеризующие координаты концов первого отрезка. Во второй строке —  $x_3, y_3, x_4, y_4$ , описывающие второй отрезок. Все координаты по модулю не превосходят 1000. Каждый отрезок параллелен одной из осей координат. То есть гарантируется, что у каждого отрезка или координаты  $x$  его концов или координаты  $y$  концов совпадают.

Также гарантируется, что отрезки невырождены, и что они не имеют общих внутренних точек и могут касаться только своими концами.

### Формат выходного файла

Если точку, удовлетворяющую условию задачи, и координаты которой по абсолютной величине не превосходят 5000 найти можно, то в первой строке выведите слово «YES». В этом случае во второй строке выведите координаты найденной точки. Если таких точек несколько, то выведите любую из них. Координаты могут оказаться вещественными, и их следует выводить с как можно большим числом знаков после десятичной точки. Разница соответствующих площадей должна быть не больше  $10^{-3}$ . Площадь каждого треугольника должна быть не меньше 0.1.

Если искомую точку найти невозможно, или координаты любой такой точки по модулю превышают 5000, то выведите только слово «NO».

### Примеры

photo.in	photo.out
0 0 0 4 5 2 5 3	YES 1.0000000000000000 0.0000000000000000
1 0 3 0 1 0 1 1	YES 3.000000000000 1.0000000000000000

## Задача G. И последние станут первыми

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

Командная олимпиада, о которой идёт речь в этой задаче, проходила по правилам, похожим на правила олимпиады, в которой вы сейчас участвуете. Отличие было только в одном: при ранжировании команд не учитывается штрафное время. При этом в случае равного количества решённых задач выше в таблице располагается та команда, чьё имя следует раньше в лексикографическом порядке.

Как и сегодня, за час до окончания олимпиады таблица результатов была «заморожена». Какое же было удивление участников, когда при подведении итогов выяснилось, что в окончательной таблице все команды расположились строго в обратном порядке, по отношению к порядку, зафиксированному «заморозкой». Так как в окончательной таблице результатов результаты сдачи каждой из задач участниками не отражены, то вам предлагается определить, могло ли такое быть в принципе, или это результат технического сбоя системы.

### Формат входного файла

Первая строка содержит два натуральных числа:  $N$  — количество команд и  $K$  — количество задач ( $1 \leq N, K \leq 100$ ).

Далее следуют  $N$  строк, описывающих саму таблицу на момент заморозки.  $i$ -я строка содержит название команды (строка из строчных латинских символов длины не более 50) и строку длины  $K$  из символов '+' и '-', показывающих успех команды по каждой из задач.

Команды упорядочены от первого места на момент заморозки к последнему.

Гарантируется, что никакие две команды не имеют одинаковое название.

### Формат выходного файла

Если порядок команд теоретически мог измениться за последний час соревнований на обратный, то сначала выведите строку «Possible», а затем таблицу окончательных результатов в формате, аналогичном таблице из входного файла (см. пример). Если соревнование так закончиться не могло, то выведите единственную строку «Impossible».

### Примеры

<code>results.in</code>	<code>results.out</code>
<code>3 4 winner +++ middle ++ looser</code>	<code>Possible looser ++++ middle +++ winner +++</code>
<code>3 4 first +++ second ++ third</code>	<code>Impossible</code>

## Задача Н. Бег по кругу

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

Спортивный программист для достижения вершин своего мастерства должен быть натренирован в совершенно разных аспектах, в том числе и физически. Кто-то для этого садится на велосипед, кто-то ныряет в бассейн, а молодой программист Влад бегают по стадиону. Но из-за неаккуратного обращения с личными вещами его секундомер может измерять время только в минутах, без указания секунд и тем более их долей. Причём, если секундомер показывает, например, 1, то это может обозначать и время ровно 2 минуты, так как  $1.(9) = 2$ .

Чтобы следить за прогрессом своего ученика, тренеру Влада приходится довольствоваться показаниями этого прибора. Каждый раз, когда Влад пробегает мимо тренера, сделав очередной круг по стадиону, тот записывает в блокнот показания секундомера в минутах. Первый раз тренер записывает время в момент, когда Влад пробежал первый круг. Фактически показания секундомера соответствуют целому числу минут, прошедших к определенному моменту времени.

На контрольной тренировке Влад бегал с постоянной скоростью, однако по записям тренера не так легко сказать, с какой именно. Напишите программу, которая поможет тренеру определить за какое минимальное, а также максимальное возможное время Влад мог пробегать каждый круг. Известно, что Влад начал бегать в момент времени 0 в том месте, где стоит тренер.

### Формат входного файла

В первой строке входного файла находится единственное натуральное число  $N$  — количество записей в блокноте тренера ( $2 \leq N \leq 10^5$ ). В следующей строке находятся сами эти записи — разделённые пробелами целые числа  $a_1, a_2, \dots, a_N$  ( $0 \leq a_1 \leq a_2 \leq \dots \leq a_N \leq 10^6$ ).

### Формат выходного файла

Выведите два неотрицательных вещественных числа, разделённых пробелом, — минимальное и максимальное возможное количество минут, за которое спортсмен пробегает один круг. Ваш ответ должен отличаться от правильного менее чем на  $10^{-3}$ .

Если ответа не существует, то есть спортсмен не мог бежать с постоянной скоростью так, чтобы записи тренера получились именно такими, в единственной строке выведите «No solution».

### Примеры

<code>run.in</code>	<code>run.out</code>
4 1 3 4 6	1.5 1.666666667
4 0 0 0 0	0 0.25
5 2 4 5 7 9	2 2
3 1 5 7	No solution

### Note

Во втором примере минимальное время может быть как угодно маленьким, и ответ 0 по сути это символизирует, но можно также считать, что Влад просто стоял на месте. Максимальный ответ в этом тесте соответствует реальным показаниям секундомера 0.25, 0.5, 0.75, 0.(9).



## Задача I. Экономия на рейсах

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

Как вы помните, Джонни работает в министерстве финансов одной небольшой страны. По роду службы ему приходится инспектировать отечественные авиакомпании и изучать маршруты, которые те предлагают.

В стране есть  $n$  городов. По государственным законам авиакомпания должна предоставлять услуги двустороннего перелёта между некоторыми парами городов, причём в целях унификации и стандартизации продолжительность любого полёта должна составлять ровно один день.

Авиакомпания называется *крупной*, если с помощью рейсов этой компании можно добраться из любого города страны до любого другого. *Крупная* авиакомпания называется *экономной*, если при этом количество маршрутов, ею предлагаемое, минимально возможное, которое может быть у *крупной* компании.

Государственная антимонопольная служба возбудила расследование против *крупной* авиакомпании «Aero-float». Ей предъявлено обвинение в излишней неэкономности. Джонни было поручено проинспектировать «Aero-float» в целях обнаружения финансовых махинаций, но авиакомпания отказалась раскрывать полный список предоставляемых ею прямых рейсов. После продолжительных переговоров руководство компании согласилось сообщить Джонни информацию о минимально возможном количестве перелётов между каждой парой городов, если использовать только прямые рейсы «Aero-float».

Используя эту информацию, помогите Джонни установить, может ли компания «Aero-float» быть *экономной* или нет? Более формально, установите, существует ли набор рейсов, при котором число рейсов в кратчайших маршрутах именно такие, как было сообщено Джонни, и при котором компания является *экономной*?

### Формат входного файла

В первой строке содержится единственное целое число  $n$  ( $2 \leq n \leq 250$ ) — количество городов в стране.

Далее идёт  $n$  строк, содержащих по  $n$  целых чисел  $T_{i,j}$ :  $j$ -е число в  $i$ -й строке равняется минимальному количеству перелётов, требуемому на перемещение из  $i$ -го города в  $j$ -й.

Гарантируется, что предоставленная информация корректна, то есть существует некоторый набор рейсов, который соответствует данному набору чисел  $T_{i,j}$ .

### Формат выходного файла

Выведите «YES», если компания «Aerofloat» может являться *экономной*, или «NO» в противном случае.

### Примеры

dist.in	dist.out
5 0 2 1 1 1 2 0 1 3 3 1 1 0 2 2 1 3 2 0 2 1 3 2 2 0	YES
4 0 1 1 1 1 0 2 2 1 2 0 1 1 2 1 0	NO

## Note

Для первого примера ниже приведён возможный вариант предлагаемых компанией маршрутов, при котором она является *экономной*. На рисунке соединены те пары городов, между которыми есть прямые рейсы.

