

Задача 1А. Летнее чтение

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Рома и Саша — одноклассники, и учитель литературы задала им на лето список литературы для прочтения. Рома и Саша всё лето будут в ЛКШ и планируют взять книги с собой. Список литературы у них одинаковый, поэтому они возьмут только по одной книге из списка на двоих.

Всего в списке n книг и прочтение i -й из них занимает t_i минут. Рома и Саша могут читать разные книги одновременно, но не могут читать одну и ту же книгу в один момент времени. Они оба хотят прочитать каждую книгу от начала до конца, не делая перерывов посередине книги. За какое минимальное время они смогут это сделать?

Формат входных данных

Первая строка содержит одно целое число n ($1 \leq n \leq 2 \cdot 10^5$) — количество книг, которые надо прочитать. Вторая строка содержит n целых чисел t_1, t_2, \dots, t_n ($1 \leq t_i \leq 10^9$) — время, необходимое для прочтения каждой книги.

Формат выходных данных

Выведите одно целое число — минимальное время, за которое Рома и Саша смогут прочитать все книги.

Пример

стандартный ввод	стандартный вывод
3 2 8 3	16

Задача 1В. Собери их все!

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

На дороге в ряд находятся n покемонов с различными номерами от 1 до n .

Егор хочет собрать их всех в порядке от 1 до n за несколько этапов. На каждом этапе Егор проходит мимо всех покемонов слева направо и собирает наибольшее возможное количество покемонов. Сколько этапов ему потребуется?

Формат входных данных

Первая строка содержит одно целое число n ($1 \leq n \leq 2 \cdot 10^5$) — количество покемонов. В следующей строке вводятся n различных чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$) — типы покемонов.

Формат выходных данных

Выведите единственное целое число — минимальное количество этапов, за которое Егор может собрать их всех.

Пример

стандартный ввод	стандартный вывод
3 1 3 2	2

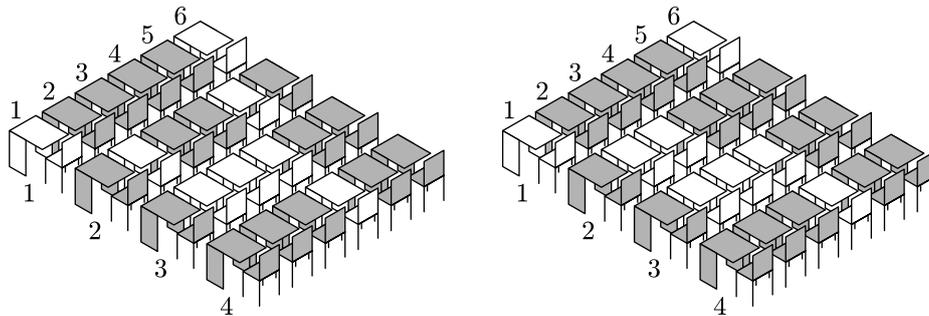
Задача 2А. Рассадка в 104й

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Александр очень любит, когда дети посещают его занятия. На каждом занятии он считает заполненность аудитории, и, если она мала, огорчается и вечером ему не хватает сил и мотивации готовиться к следующему дню. Дети не хотят расстраивать своего любимого преподавателя, поэтому очень хотят максимизировать заполненность аудитории.

Класс представляет собой прямоугольник $w \times h$. В каждой ячейке прямоугольника находится место для одного ученика. Заполненность **аудитории** — это минимум из заполненности строк и заполненности столбцов. Заполненность **строк** — это максимальное число учеников, сидящих в одной строке. Заполненность **столбцов** — это максимальное число учеников, сидящих в одном столбце.

К примеру, на картинке слева нарисована рассадка 16 учеников (занятые учениками парты нарисованы темным). Заполненность строк в этой рассадке равна 5 (достигается в четвертой строке) и заполненность столбцов равна 3 (достигается в первом, третьем, пятом и шестом столбцах). Тем самым заполненность класса равна 3. Но если ученики перераспределятся так, как показано на правой картинке, то заполненность столбцов станет равна 4 (в пятом столбце), и заполненность класса станет равна 4.



Ученики Александра хотят распределиться так, чтобы заполненность класса была максимальной. Помогите им написать программу.

Формат входных данных

Первая строка содержит три целых числа n , w и h — число учеников, строк и столбцов в классе ($1 \leq w, h \leq 100$, $1 \leq n \leq w \times h$).

Формат выходных данных

В первой строке выведите одно целое число — максимальную заполненность класса.

в следующих w строках выведите рассадку учеников. Каждая строка должна состоять из h символов ‘.’ или ‘#’, ‘.’ обозначает пустую парту, а ‘#’ — занятую.

Если есть несколько рассадок с максимальной заполненностью класса, можно вывести любую.

Пример

стандартный ввод	стандартный вывод
16 4 6	4 .####. #..### #...## ###.##

Задача 2В. Чебурашка

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

По кругу расположено n ящиков с апельсинами, пронумерованных от 0 до $n - 1$, в i -м ящике лежит a_i апельсинов.

Чебурашка несколько раз (возможно, ноль) повторяет следующее действие:

Он выбирает число $0 \leq i \leq n - 1$, после чего для каждого j от 1 до n съедает из $((i + j) \bmod n)$ -го ящика j апельсинов (в случае, если в каком-то ящике апельсинов меньше, чем он планирует съесть, он не может выбрать такое i).

Помогите Чебурашке: скажите ему, сможет ли он съесть все апельсины.

Формат входных данных

На первой строке дано число n ($1 \leq n \leq 10^5$) — число ящиков.

Во второй строке находятся числа a_0, a_1, \dots, a_{n-1} ($1 \leq a_i \leq 10^9$) — сколько апельсинов в каждом из ящиков.

Формат выходных данных

Выведите «YES» (без кавычек), если Чебурашка может съесть все апельсины, и «NO» в противном случае.

Примеры

стандартный ввод	стандартный вывод
5 2 3 4 5 1	YES
5 10 8 6 9 12	YES
4 2 3 1 1	NO

Задача 3А. Ладьи на доске

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Даша любит играть в шахматы. У нее есть шахматная доска необычного размера — $n \times m$. Ее любимая фигура — ладья.

Еще, она любит решать разные головоломки. Одна из последних, с которой она столкнулась, как раз про ладьи. У Даши есть $n \cdot m$ одинаковых ладей и ее необычная доска. Она хочет расставить все ладьи на доске одна за одной, заполнив все клетки. Единственное ограничение, каждый раз, когда она ставит ладью, новая ладья должна находиться под боем у **четного** количества уже поставленных.

Помогите Даше решить головоломку. Даны n и m , найдите порядок клеток, на которые Даша должна выставлять ладьи.

Формат входных данных

В единственной строке содержатся два целых числа n и m ($1 \leq n, m \leq 500$).

Формат выходных данных

Если существует искомый порядок клеток, первая строка должна содержать «YES», а следующие $n \cdot m$ строк должны содержать пары целых чисел x_i и y_i ($1 \leq x_i \leq n, 1 \leq y_i \leq m$) — координаты клеток.

Если не решения не существует, выведите «NO».

Примеры

стандартный ввод	стандартный вывод
1 1	YES 1 1
1 2	NO

Задача 3В. Угадайка

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Как известно, в алгоритмических параллелях ЛКШ каждый день читается лекция на определённую тему. Школьники же, будучи очень любопытными, всегда пытаются узнать тему заранее. И вот однажды одному из преподавателей надоело постоянно от них отшучиваться, и он решил дать ученикам своей параллели информацию сразу про все дни смены! Но не просто так, а в игровой форме.

В тот год в той параллели было n учебных дней и n тем, которые школьникам были известны заранее, поэтому расписание смены можно представить как перестановку a_1, a_2, \dots, a_n чисел от 1 до n , которую знает только преподаватель. Школьникам разрешается задать не более 10 вопросов. Каждый вопрос имеет вид бинарной строки $\overline{q_1 q_2 \dots q_n}$ из символов 0 и 1. В ответ преподаватель выдаст им строку $\overline{q_{a_1} q_{a_2} \dots q_{a_n}}$.

Помогите школьникам за отведённое количество вопросов узнать расписание смены.

Протокол взаимодействия

Это интерактивная задача. Ваша программа должна будет задать не более 10 вопросов программе жюри, а затем вывести ответ.

Сначала вашей программе подаётся на вход число n ($1 \leq n \leq 1000$) — количество дней в смене.

Далее ваша программа начинает задавать вопросы. Каждый вопрос имеет вид $? q_1 q_2 \dots q_n$ и означает, что школьники спросили у преподавателя про строку $\overline{q_1 q_2 \dots q_n}$. В ответ программа жюри выведет строку $\overline{r_1 r_2 \dots r_n}$ — ответ на вопрос. **Вам разрешается задать не более 10 вопросов.**

Как только ваша программа нашла ответ, она должна вывести его в формате $! a_1 a_2 \dots a_n$, где a_i образуют перестановку чисел от 1 до n — искомое расписание, **и сразу же завершиться.**

Не забывайте переводить строку и сбрасывать буфер после каждого вывода (команда `cout << endl` в C++).

Пример

стандартный ввод	стандартный вывод
4	? 1000
1000	? 0100
0010	? 0010
0001	! 1 4 2 3

Задача 4А. Найди произведение

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Коля построил две бесконечные арифметические прогрессии A и B . Первая последовательность выглядит так: $A = [a, a + p, a + 2 \cdot p, a + 3 \cdot p, \dots]$, вторая последовательность выглядит так: $B = [b, b + q, b + 2 \cdot q, b + 3 \cdot q, \dots]$.

Например, если $a = 2, p = 3, b = 3, q = 9$, то последовательности выглядят так:

$$A = [2, 5, 8, 11, \dots]$$

$$B = [3, 12, 21, 30, \dots]$$

Теперь Коля хочет выбрать по одному числу из каждой последовательности так, чтобы их произведение было равно X . Посчитайте, сколькими способами он может это сделать.

Например, если $X = 24$, то для последовательностей выше можно выбрать числа двумя способами: 2 из A и 12 из B , или 8 из A и 3 из B .

Формат входных данных

Первая строка ввода содержит числа a, p, b, q и X ($1 \leq a, p, b, q, X \leq 10^9$).

Формат выходных данных

Выведите одно число — число способов выбрать два числа, чтобы их произведение было равно X .

Примеры

стандартный ввод	стандартный вывод
2 3 3 9 24	2
2 2 4 6 239	0
1 1 1 1 1000000000	100

Задача 4В. Игра "Мемори"

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Возможно вы знаете игру "Мемори". В этой игре на стол раскладываются несколько карточек, тыльной стороной вверх. Тыльные стороны карточек выглядят одинаково. Среди карточек есть одинаковые, одинаковых карточек каждого типа всегда ровно две.

Ход в игре происходит следующим образом: игрок открывает одну из карточек, видит, что на ней нарисовано, затем открывает вторую карточку. Если вторая карточка такая же, как первая, то игрок забирает обе карточки себе и продолжает ходить. Если карточки разные, то ход игрока заканчивается.

Сейчас уже прошло несколько ходов игры, и вы запомнили про некоторые карточки, что на них нарисовано. Сколько пар карточек вы гарантированно можете забрать за один ход?

Формат входных данных

Первая строка содержит четное число n — число карточек, которые остались на столе ($2 \leq n \leq 50$). Вторая строка содержит строку из n символов, которая описывает карточки. Карточки, про которые вы помните, что на них нарисовано, обозначены буквами, одинаковые буквы соответствуют одинаковым карточкам. Карточки, про которые вы не знаете, что на них нарисовано, обозначены символом «?».

Формат выходных данных

Выведите одно число — максимальное число пар карточек, которые вы гарантированно забрать за свой ход.

Примеры

стандартный ввод	стандартный вывод
8 A?AC??CT	2
4 AB??	2
2 ??	1
4 X???	0

Задача 5А. Красивая перестановка

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Перестановка целых чисел $1, 2, \dots, n$ называется красивой, если нет соседних элементов, разность которых равна 1. Постройте лексикографически минимальную красивую перестановку длины n , если такая перестановка существует.

Формат входных данных

Единственная строка содержит целое число n ($1 \leq n \leq 10^6$).

Формат выходных данных

Выведите лексикографически минимальную красивую перестановку целых чисел $1, 2, \dots, n$. Если такой перестановки нет, выведите «NO SOLUTION».

Примеры

стандартный ввод	стандартный вывод
5	1 3 5 2 4
3	NO SOLUTION

Задача 5В. Хомячки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Для изучения поведения хомячков ученые построили кольцевой тоннель длиной l метров. В этот тоннель запустили n хомячков. Позиция хомячка задается целым числом от 0 до $l - 1$, соответствующим расстоянию, которое нужно пройти по тоннелю от выделенной точки до хомяка по часовой стрелке. Каждый хомячок начинает двигаться по тоннелю в каком-то направлении со скоростью 1 метр в минуту. Когда два хомячка встречаются, они разворачиваются и начинают двигаться в противоположном направлении с той же скоростью.

Вам известны начальное положение и направление движения каждого хомячка. Ваша задача — узнать их положение через t минут.

Формат входных данных

Первая строка содержит числа n , l и t ($1 \leq n \leq 1000$, $n \leq l \leq 10^9$, $1 \leq t \leq 10^9$). Следующие n строк содержат описание начального состояния хомячков. Каждое такое описание состоит из начального положения хомячка x_i ($0 \leq x_i \leq l - 1$) и его направления. Число 1 соответствует направлению по часовой стрелке, а число -1 — против часовой стрелки.

Хомячки упорядочены по возрастанию позиции x_i , все позиции различны.

Формат выходных данных

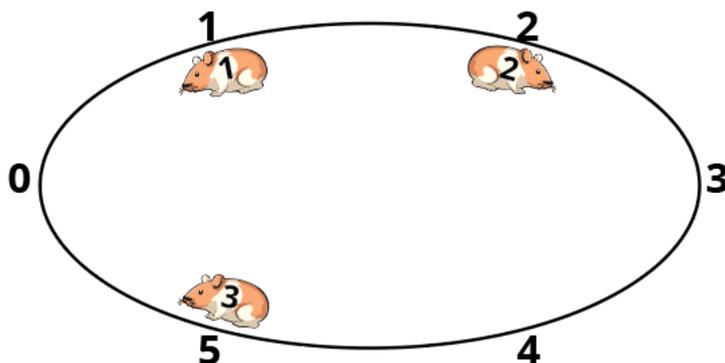
Выведите n чисел — положение каждого хомячка через t минут.

Примеры

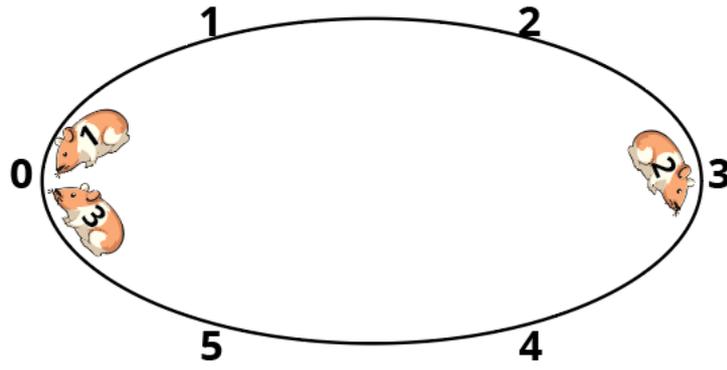
стандартный ввод	стандартный вывод
3 6 3 1 -1 2 1 5 1	2 4 5
2 3 1 0 1 2 -1	1 1
5 5 42 0 1 1 1 2 1 3 1 4 1	2 3 4 0 1

Замечание

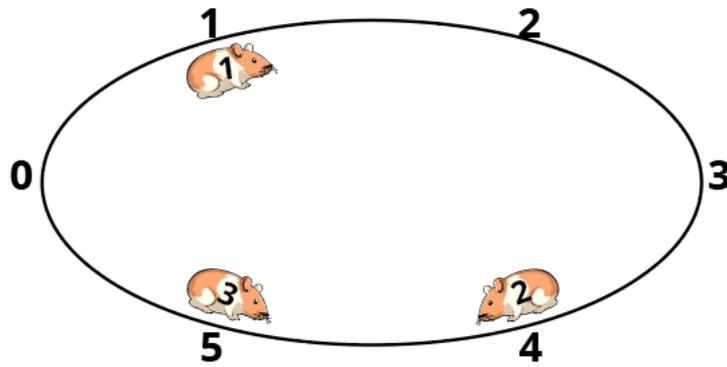
В первом примере начальное положение хомячков такое.



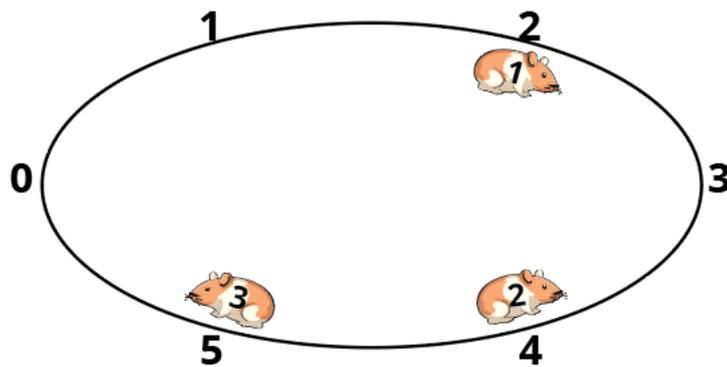
Через одну минуту оно будет такое.



Хомячки 1 и 3 встретились, и теперь побегут в противоположные стороны. Через две минуты состояние будет такое:



Через две с половиной минуты хомячки 2 и 3 встретятся в точке 4.5 и побегут обратно, поэтому через три минуты состояние будет такое:



Задача 6А. Упрощенное скрещивание

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Биоинформатик Федот решил заняться скрещиванием двух организмов. Последовательности ДНК этих организмов — это две строки s_0 и t_0 , состоящие из символов ‘А’, ‘С’, ‘G’ и ‘Т’.

Поскольку Федоту всего лишь 5 лет, он не очень знает, как происходит скрещивание, но абсолютно уверен, что если у него есть последовательности s_i и t_i , это происходит следующим образом:

- Строка s_i делится на две строки (возможно нулевой длины) s_i^p и s_i^s такие, что их конкатенация $s_i^p + s_i^s$ равна s_i (то есть s_i^p — префикс строки s_i , а s_i^s — оставшийся суффикс)
- Аналогичным образом t_i делится на t_i^p и t_i^s
- Исходные последовательности (строки) заменяются на $s_{i+1} = s_i^p + t_i^s$ и $t_{i+1} = t_i^p + s_i^s$

Федот хочет произвести скрещивание несколько раз, чтобы в конце получить две последовательности ДНК s_k и t_k такие, что s_k состоит только из аденина (‘А’), а t_k — не содержит аденин вообще. Помогите ему добиться этого за минимальное число скрещиваний.

Формат входных данных

Ввод состоит из двух строк — s_0 и t_0 — последовательностей ДНК двух исходных организмов ($1 \leq |s_0|, |t_0| \leq 10^6$).

Формат выходных данных

В первой строке выведите число k — минимальное количество шагов скрещивания, которое необходимо для переноса всего аденина в первый организм.

В i -й (нумеруя с нуля) из следующих k строк выведите через пробел $|s_i^p|$ и $|t_i^p|$ — длины выбранных префиксов последовательностей на i -м шагу скрещивания.

Пример

стандартный ввод	стандартный вывод
ACAT	3
GAA	3 1
	2 0
	1 0

Замечание

В примере из условия скрещивания происходят следующим образом:

1. «ACAT» делится на «ACA» и «Т», и «GAA» делится на «G» и «AA». После обмена суффиксами получаются «АСААА» и «GT».
2. Затем «АСААА» делится на «АС» и «ААА», тогда как «GT» делится на пустой префикс «» и суффикс «GT». После обмена получаются «ACGT» и «ААА».
3. Последним шагом отделяются префиксы длин 1 («А») и 0 («»), и в результате строки оказываются равны «АААА» и «CGT».

Можно доказать, что за два скрещивания желаемого результата добиться нельзя.

Задача 6В. Препятствующая полоса

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Вышедший на пенсию бегун Усейн Олд решил в качестве утренней тренировки пройти полосу препятствий. Полоса представляет из себя последовательность из n столбиков различной целой высоты. В каждый момент времени Усейн находится на каком-то из столбиков и может переместиться только на соседний с ним.

Усейн уже не молод, поэтому не хочет повредить суставы при прохождении полосы. Перемещение со столбика высоты h на соседний называется *безопасным*, если его высота находится между $h - 3$ и $h + 2$ включительно. Если высота нового столбика хотя бы на 4 ниже, то при прыжке есть риск повредить ноги, а если хотя бы на 3 выше — при подъеме можно повредить руки. Также обратите внимание на то, что Усейн может двигаться как в правую, так и в левую сторону.

Помогите Усейну найти наиболее длинный участок полосы препятствий, который он может безопасно преодолеть. А именно, подберите пару чисел s, t с максимальным $|s - t|$, чтобы Усейн мог добраться от s -го столбика до t -го, производя лишь безопасные перемещения.

Формат входных данных

В первой строке ввода через пробел заданы два целых числа m и n — количество следующих строк во вводе и количество столбиков ($1 \leq n, m \leq n \cdot m \leq 10^6$).

Следующие m строк задают полосу препятствий в виде матрицы размера $m \times n$. Каждая строка имеет длину n и состоит из символов «.» и «#». Символ «.» означает, что данная клетка ничем не занята, а «#» — что эта клетка занята столбиком. Все столбики начинаются от уровня земли. Таким образом, если в i -й строке на j -м месте стоит решетка, то и во всех следующих строках на этой позиции будет находиться решетка. Высота каждого столбика задается количеством решеток в соответствующем столбце матрицы.

Формат выходных данных

Выведите через пробел два целых числа s и t , обозначающих границы движения Усейна по полосе. Обратите внимание, что направление движения имеет значение, и ответы «s t» и «t s» отличаются.

Примеры

стандартный ввод	стандартный вывод
5 7 #..... #...#. #...##. ##.### #####	1 7
8 6 #...#. #.#.#. #.#.#. #.#.#. #.#.## ###.## #####	1 1

Замечание

В первом примере ответ «1 7» единственный верный. При $s = 2, t = 7$ Усейн всё ещё сможет

добраться от старта до финиша, но это не два самых далёких столбика, между которыми он сумеет переместиться. При $s = 7$, $t = 1$ Усейну нужно будет перебраться со второго столбика на первый, что небезопасно.

Во втором примере между всеми парами столбиков слишком большие перепады высот, поэтому оптимально будет оставаться на месте на любом столбике.