

## Задача А. Минёр

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Каждый сапёр знает, что вероятность подорваться на mine очень сильно зависит от того, насколько профессионально они расставлены. Широко известная игра «Сапёр» проходит на прямоугольном поле из  $w \times h$  клеток. В каждой клетке либо стоит мина, либо записано количество мин, располагающихся в восьми соседних клетках (число от 0 до 8). В этой задаче вам необходимо узнать, какое минимальное количество мин потребуется поставить на поле, чтобы каждая клетка либо содержала мину, либо находилась рядом с ней.

### Формат входных данных

В первой строке входа находятся два целых числа  $w$  и  $h$  ( $1 \leq w, h \leq 1000$ ), разделённые пробелом, — ширина и высота поля соответственно.





### Формат выходных данных

Выведите минимальное количество мин, достаточное для того, чтобы в каждой клетке игрового поля либо стояла мина, либо было записано число больше нуля.

### Пример

<i>Входные данные</i>	<i>Выходные данные</i>
5 5	4

Пример (поле  $5 \times 5$ ) проиллюстрирован на следующем рисунке:

1	1	1	1	
1		1	1	1
1	1	2	1	1
1	1	2		1
1		2	1	1

## Задача В. Битовая строка

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Пусть у нас есть некоторая строка. Если мы переставим последний её символ из конца в начало, мы получим *циклический сдвиг* данной строки. Прodelывая эту операцию снова и снова, мы получим множество всех циклических сдвигов строки. Например, множество всех циклических сдвигов строки «0011» содержит строки «0011», «1001», «1100», «0110». Строка называется *простой*, если она лексикографически меньше либо равна каждому из своих циклических сдвигов. Найдите количество простых строк длины  $n$ , состоящих только из символов «0» и «1».

### Формат входных данных

На вход подаётся целое число  $n$  ( $1 \leq n \leq 10$ ).

### Формат выходных данных

Выведите одно целое число — ответ на задачу.

### Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
4	6
9	60

В первом примере простыми являются следующие строки: «0000», «0001», «0011», «0101», «0111», «1111».

## Задача С. Многогранник

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Дан выпуклый многогранник. Никакие три его вершины не лежат на одной прямой, и никакие две его грани не лежат в одной плоскости. Масса многогранника равномерно распределена по его объёму. Вам нужно найти центр тяжести этого многогранника.

### Формат входных данных

В первой строке вводится целое число  $n$  — количество граней многогранника ( $4 \leq n \leq 55$ ). Далее следуют  $n$  блоков, описывающих грани. В первой строке блока записано целое число  $m$  — количество вершин грани ( $3 \leq m \leq 10$ ). В каждой из следующих  $m$  строк блока находится по три целых числа через пробел — координаты очередной вершины в порядке обхода. Все координаты по модулю не превышают 100.

### Формат выходных данных

Выведите три числа через пробел — координаты центра тяжести. Числа следует вывести с точностью не менее трёх знаков после десятичной точки.

### Пример

<i>Входные данные</i>	<i>Выходные данные</i>
6 4 0 0 0 0 1 0 1 1 0 1 0 0 4 0 0 1 0 1 1 1 1 1 1 0 1 4 0 0 0 0 0 1 0 1 1 0 1 0 4 1 0 0 1 0 1 1 1 1 1 1 0 4 0 0 0 0 0 1 1 0 1 1 0 0 4 0 1 0 0 1 1 1 1 1 1 1 0	0.500000 0.500000 0.500000

## Задача D. Многоугольник

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Дан выпуклый многоугольник из  $n$  вершин. Никакие три вершины многоугольника не лежат на одной прямой. Соединим каждую пару несмежных вершин диагональю. Требуется найти количество пар пересекающихся диагоналей (диагонали, имеющие общую вершину, не считаются пересекающимися).

### Формат входных данных

В первой строке записано целое число  $n$  ( $3 \leq n \leq 50$ ). Далее в  $n$  строках следуют пары целых чисел — координаты вершин многоугольника в порядке обхода против часовой стрелки. Координаты по модулю не превосходят миллиона.

### Формат выходных данных

Одно целое число — ответ на задачу.

### Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
3 0 0 1 0 1 1	0
4 0 0 1 0 1 1 0 1	1

## Задача Е. Матрицы

Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Квадратная матрица  $A_{n \times n}$  называется симметрической, если  $a_{ij} = a_{ji}$  для всех  $1 \leq i, j \leq n$ .

Квадратная матрица  $A_{n \times n}$  называется кососимметрической, если  $a_{ij} = -a_{ji}$  для всех  $1 \leq i, j \leq n$ .

Пусть задана некоторая квадратная матрица  $A_{n \times n}$ . Требуется определить, можно ли представить её в виде суммы симметрической и кососимметрической матриц.

### Формат входных данных

В первой строке входа задано целое число  $n$  ( $1 \leq n \leq 200$ ). В следующих  $n$  строках содержится по  $n$  целых чисел — элементы матрицы  $a_{ij}$  ( $-1\,000 \leq a_{ij} \leq 1\,000$ ).

### Формат выходных данных

В первой строке выведите «YES», если матрицу можно разложить указанным образом. Далее в  $n$  строках запишите по  $n$  чисел — элементы полученной симметрической матрицы с не менее чем тремя знаками после десятичной точки. Затем выведите элементы полученной кососимметрической матрицы в аналогичном формате. Выведенные матрицы должны быть разделены одной пустой строкой. Если существует несколько разложений, выведите любое.

Если разложение невозможно, выведите в первой строке «NO».

### Пример

<i>Входные данные</i>	<i>Выходные данные</i>
3	YES
2 -2 6	2.0000000 -1.0000000 3.0000000
0 0 0	-1.0000000 0.0000000 5.0000000
0 10 -4	3.0000000 5.0000000 -4.0000000
	0.0000000 -1.0000000 3.0000000
	1.0000000 0.0000000 -5.0000000
	-3.0000000 5.0000000 0.0000000

## Задача F. Точки

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Дано множество точек на плоскости. Покрасим каждую из этих точек в белый либо чёрный цвет и рассмотрим прямую, которая не проходит ни через одну из них. Будем называть такую прямую разделяющей, если не существует пары точек разных цветов, лежащих по одну сторону от неё. Требуется определить, существует ли такой способ покрасить точки, что для них не будет существовать разделяющей прямой.

### Формат входных данных

В первой строке входа находится целое число  $n$  — количество точек ( $2 \leq n \leq 100$ ). Далее в  $n$  строках следуют по два целых числа в каждой — координаты точек. Все координаты по модулю не превосходят 10 000. Гарантируется, что все точки различны.

### Формат выходных данных

Выведите «YES», если искомая раскраска существует, и «NO» в противном случае.

### Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
2 1 1 2 4	NO
3 1 1 2 4 3 9	NO

## Задача G. Очередная система уравнений

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Задана матрица  $(f_{ij})_{m \times n}$ , числа  $b_i$  ( $1 \leq i \leq m$ ) и  $d_j$  ( $1 \leq j \leq n$ ), причём  $\sum_{i=1}^m b_i = \sum_{j=1}^n d_j$ . Требуется отыскать решение системы из  $n + m$  линейных уравнений следующего вида:

$$\sum_{j=1}^n f_{ij} y_{ij} = b_i \quad (\text{для любого } i = 1, 2, \dots, m),$$
$$\sum_{i=1}^m f_{ij} y_{ij} = d_j \quad (\text{для любого } j = 1, 2, \dots, n),$$

удовлетворяющее ограничению  $0 \leq y_{ij} \leq 10^6$ .

### Формат входных данных

На входе в первой строке записаны целые числа  $m$  и  $n$  — размерности системы ( $1 \leq m, n \leq 50$ ). Во второй и третьей строках записаны целые числа  $b_i$  ( $i = 1, 2, \dots, m$ ) и  $d_j$  ( $j = 1, 2, \dots, n$ ) соответственно ( $1 \leq b_i, d_j \leq 10^6$ ). Остальные  $m$  строк по  $n$  чисел образуют матрицу системы  $(f_{ij})$ . Числа  $f_{ij}$  целые,  $1 \leq f_{ij} \leq 100$ .

### Формат выходных данных

Если указанная система линейных уравнений не имеет решения, ответом будет строка «Solution is not found.».

Если же решение существует, то в первой строке нужно вывести фразу «Solution is found:», в остальных  $m$  строках — матрицу переменных  $(y_{ij})$ , удовлетворяющую системе. Если решений несколько, то в качестве ответа можно выбрать любое.

Все переменные должны быть выведены с точностью не менее восьми знаков после десятичной точки. Формат вывода обозначен в примерах ниже.

### Примеры

Входные данные	Выходные данные
2 2 3 4 2 5 4 7 2 4	Solution is found: 0.50000000 0.14285714 0.00000000 1.00000000
2 2 7 3 2 8 2 3 4 1	Solution is found: 1.00000000 1.66666667 0.00000000 3.00000000
2 3 10 20 5 15 10 1 1 1 1 1 1	Solution is found: 5.00000000 5.00000000 0.00000000 0.00000000 10.00000000 10.00000000

## Задача Н. Число Первого Тренера

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Ладья на шахматной доске может перемещаться по вертикали или горизонтали на любое ненулевое расстояние, не выходя за границы доски.

Летопись АСМ гласит, что Первый Тренер поставил ладью на шахматную доску размером  $n \times n$ . Затем он сделал четыре хода. Известно, что последним ходом он вернул ладью в исходную клетку. Также известно, что при этом ладья побывала в четырёх различных точках. Остальное летопись утаивает.

Хранитель традиций называет количество способов, которыми Первый Тренер мог это сделать, — Числом Первого Тренера.

### Формат входных данных

Во входе записано целое число  $n$  ( $2 \leq n \leq 50$ ).

### Формат выходных данных

Выведите Число Первого Тренера.

### Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
2	8
3	72

В первом примере возможны следующие способы поставить ладью и сделать четыре хода:

1. a1–a2–b2–b1–a1,
2. a1–b1–b2–a2–a1,
3. a2–a1–b1–b2–a2,
4. a2–b1–b1–a1–a2,
5. b1–a1–a2–b2–b1,
6. b1–b2–a2–a1–b1,
7. b2–a2–a1–b1–b2,
8. b2–b1–a1–a2–b2.



## Задача I. Битовая сортировка

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Отсортируем все целые числа от 0 до  $n$  включительно по количеству единиц в их двоичном представлении. Таким образом, число  $4 = 100_2$  идёт раньше числа  $3 = 11_2$ , так как в двоичном представлении имеет на одну единицу меньше. В случае одинакового количества единиц раньше идёт меньшее число.

Пример отсортированных чисел для  $n = 10$ : 0, 1, 2, 4, 8, 3, 5, 6, 9, 10, 7.

Даны числа  $n$  и  $k$ . Требуется найти следующее после  $k$  число в указанном выше порядке.

### Формат входных данных

В первой строке записано целое число  $n$  ( $1 \leq n < 10^{100}$ ). Вторая строка содержит целое число  $k$  ( $0 \leq k \leq n$ ).

### Формат выходных данных

На выход выведите следующее за  $k$  число. В случае если  $k$  — последнее в отсортированной последовательности число, выведите «-1».

### Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
10 7	-1
123 0	1
51 44	49

## Задача J. Путешествие короля

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Шахматный король стоит на клетке бесконечной шахматной доски. За ход он может по обычным шахматным правилам перейти на одну клетку по горизонтали, вертикали или диагонали. Сколько существует различных путей короля заданной длины, которые закончатся в исходной клетке? При этом исходную клетку можно посетить только в начале и в конце пути. Остальные клетки можно посещать сколько угодно раз.

### Формат входных данных

На вход подаётся целое число  $l$  — количество ходов короля ( $1 \leq l \leq 12$ ).

### Формат выходных данных

Выведите количество различных маршрутов короля, состоящих ровно из  $l$  ходов.

### Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
1	0
2	8

## Задача К. Хранитель традиций

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Хранитель традиций загадал целое положительное число. Про него известно, что при делении на 2 и 3 оно даёт остаток 1, при делении на 5 — остаток 2, при этом оно делится на 7. Конечно, по этим данным не угадаешь число. Но найдите хотя бы минимальное из всех возможных.

### Формат входных данных

На вход подаётся пустая строка.

### Формат выходных данных

Одно целое положительное число — ответ на задачу.

### Пример

Для этой задачи нет примера.

## Задача L. Дроби

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Дано целое число  $n$ . Найти количество несократимых обыкновенных дробей, меньших единицы, числитель и знаменатель которых положительны, при этом знаменатель меньше либо равен  $n$ .

### Формат входных данных

На вход подаётся целое число  $n$  ( $1 \leq n \leq 10$ ).

### Формат выходных данных

Одно целое число — ответ на задачу.

### Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
1	0
2	1
3	3