Задача 1. Дом семьи Гарнетт

Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Совсем недавно семья Гарнетт переехала в новый город. Земельный участок, приобретенный ими, находится рядом с домом, в котором живут Шерлок Холмс и доктор Ватсон. Гарнетты рады такому знакомству, ведь теперь в случае загадочных и необъяснимых происшествий у них будет возможность посоветоваться со знаметитым сыщиком.

Однако, в проблеме, стоящей перед Гарнеттами сейчас, нет ничего загадочного. Участок, на котором они планируют строить себе дом, имеет форму прямоугольника размеров $n \times m$ метров. Гарнетты, как и все истинные англичане, любят порядок, и поэтому они хотят, чтобы их дом также имел прямоугольную форму, его стены были параллельны сторонам участка, а расстояние от любой стены дома до параллельной ей границы участка было бы целым числом метров. При этом они, конечно же, хотят построить дом максимальной площади.

К сожалению, есть проблема, мешающая им построить дом, совпадающий границами с участком. Заключается она в том, что на участке расположены две скважины с водой, и Гарнетты хотят, чтобы одна из них оказалась внутри дома, а другая — за его пределами. Чтобы узнать максимальную площадь, которую может иметь дом, отвечающий описанным выше требованиям, они обратились к Шерлоку Холмсу. Помогите ему ответить им на этот вопрос.

Формат входных данных

Для удобства разобьем участок на $n \times m$ квадратов единичной площади. Каждая из скважин находится ровно в одном квадрате и полностью его занимает. Скважины находятся в разных квадратах. Вершины дома совпадают с вершинами квадратов.

В первой строке даны два числа n и m — размеры участка ($2 \le n, m \le 1000$). Каждая из n последующих строк содержит по m чисел — описания квадратов единичной площади. Если в данном квадрате расположена скважина, то соответствующее число равно единице, иначе число равно нулю.

Гарантируется, что на участке ровно две скважины (ровно два числа, описывающих квадраты единичной площади, равны единице).

Формат выходных данных

Выведите одно число — максимальную площадь дома, отвечающего всем требованиям

Пример

Стандартный ввод	Стандартный вывод
3 4 0 1 0 0 0 0 0 0 0 0 0 1	9

Подзадача 1 (30 баллов)

n и m не превышают 10. Тесты 2-15. Баллы за подзадачу начисляются только в случае, если все тесты успешно пройдены.

Подзадача 2 (30 баллов)

n и m не превышают 70. Тесты 16-30. Баллы за подзадачу начисляются только в случае, если все тесты этой и предыдущей группы успешно пройдены.

Подзадача 3 (40 баллов)

Дополнительные ограничения отсутствуют. Тесты 31-50. Баллы за подзадачу начисляются только в случае, если все тесты этой и предыдущих групп успешно пройдены.

Получение информации о результатах окончательной проверки

По запросу сообщается результат окончательной проверки на каждом тесте.

Задача 2. Марсианский дом

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Марсиане живут в круглых домах. Рассмотрим один из таких домов. Он разделён на **N** комнат, пронумерованных целыми числами от 1 до **N** по часовой стрелке, причём **i**-я комната граничит с **i**+1-й для всех $1 \le \mathbf{i} \le \mathbf{N}$-1. Также первая комната граничит с последней комнатой. В каждой комнате живет ровно один марсианин, в **i**-й комнате живет марсианин с номером **i**.

Марсиане – существа дружелюбные, но бывает и так, что они враждуют друг с другом. Введём для каждого марсианина коэффициент его *дружелюбности*. *Дружелюбность***i**-го марсианина равна $\mathbf{A_i}$. Известно, что два марсианина с дружелюбностью **X** и **Y** соответственно враждуют, если $\mathbf{X} \& \mathbf{Y} \neq 0$. Здесь & обозначает операцию побитового логического И.

В одном доме могут жить враждующие между собой марсиане. В таких случаях строят некоторое число перегородок между комнатами, разделяя дом на несколько квартир. Марсиане при этом стараются разделять дома на как можно меньшее число квартир так, чтобы никакие два марсианина в одной квартире не враждовали. Заметим, что в доме может быть всего одна квартира. В этом случае необходимо построить ровно одну перегородку в любом месте.

Строители Олимп-Сити сейчас как раз строят один из домов. Они уже знают, кто будет жить в этом доме. Им необходимо определить, каким образом разбить дом на квартиры так, чтобы никакие два марсианина в одной квартире не враждовали, и при этом квартир было как можно меньше. Помогите им найти ответ на этот вопрос

Формат входных данных

Первая строка содержит одно целое число **N** ($1 \le \mathbf{N} \le 200\,000$) – количество комнат в доме.

Во второй строке записаны**N** целых чисел $\mathbf{A_i}$($1 \le \mathbf{A_i} \le 10^9$) – дружелюбностимарсиан.

Формат выходных данных

В первой строке выведите два целых числа **K** и **S** ($1 \le \mathbf{K}, \mathbf{S} \le \mathbf{N}$) – минимально возможное количество квартир в доме и номер комнаты, левее которой необходимо построить одну из перегородок.

В следующей строке выведите **K** целых чисел $\mathbf{B_i}$ ($1 \le \mathbf{B_i} \le \mathbf{N}$) – количество комнат в **i**-й квартире. Квартиры нумеруются по часовой стрелке, начиная с комнаты с номером **S**. Заметим, что должно выполняться $\mathbf{B_1} + \mathbf{B_2} + \ldots + \mathbf{B_K} = \mathbf{N}$.

Если существует несколько оптимальных разбиений, разрешается вывести любое из них.

Примеры

Стандартный ввод	Стандартный вывод	Комментарий
7 1 1 2 12 8 6 16	2 2 3 4	Поставив перегородки между комнатами 1 и 2, а также между комнатами 4 и 5, мы получим требуемое разбиение. Комната 1 окажется в квартире с комнатами 5, 6 и 7
5 1 2 1 2 1	3 1 2 2 1	Первая квартира состоит из первых двух комнат, вторая – из комнат 3 и 4, третья – из комнаты 5.
6 1 2 3 4 5 6	4 6 2 1 2 1	Перегородки можно расставить, например, следующим образом: 1 | 2 | 3 4 | 5 | 6 Комнаты 1 и 2 тоже можно было объединить в одну квартиру, но число квартир от этого не уменьшится

Система оценки и описание подзадач

В этой задаче, кроме тестов 1–3 из условия, 5 подзадач. Баллы за каждую подзадачу начисляются только в случае, если все тесты этой и предыдущих подзадач успешно пройдены.

Подзадача 1 (20 баллов)

$N \le 50$, $A_i \le 2$
В этой подзадаче 10 тестов (с 4 по 13).

Подзадача 2 (20 баллов)

$N \le 50$, A_i являются степенью двойки.
В этой подзадаче 10 тестов (с 14 по 23).

Подзадача 3 (20 баллов)

$N \le 50$.
В этой подзадаче 10 тестов (с 24 по 33).

Подзадача 4 (20 баллов)

$N \le 300$.
В этой подзадаче 10 тестов (с 34 по 43).

Подзадача 5 (20 баллов)

Нет дополнительных ограничений.
В этой подзадаче 10 тестов (с 44 по 53).

Получение информации о результатах окончательной проверки

Сообщается результат окончательной проверки на каждом тесте.

Задача 3. Единственная задача в туре не про дома □

Ограничение по времени: 0.4 секунды
Ограничение по памяти: 512 мегабайт

Известная марсианская компания «Interplanetary Software, Inc.», которая специализируется на обработке строковой информации, разрабатывает новый текстовый редактор. Этот текстовый редактор работает с некоторой строкой, состоящей из строчных букв латинского алфавита. Пока что он поддерживает только две команды:

1. Добавить произвольный символ в конец строки.
2. Удалить последний символ в строке (если строка непустая).

Ещё текстовый редактор умеет проверять, насколько строки отличаются друг от друга. Назовем эту величину *непохожестью*. *Непохожесть* $\mathbf{d}(\mathbf{A}, \mathbf{B})$двух строк $\mathbf{A}$ и $\mathbf{B}$ – минимальное количество команд, необходимое, чтобы получить из строки $\mathbf{A}$ строку $\mathbf{B}$. Например, $\mathbf{d}$(`‘tests’`, `‘text’`) = 5, поскольку строку `‘tests’` можно превратить в `‘text’` за пять команд следующим образом: сначала удалить три последних символа, а затем добавить символы `‘x’` и `‘t’` в конец.

Кроме того, текстовый редактор должен уметь считать величину *уникальности* набора строк. Эта величина считается следующим образом. Пусть у нас есть набор из $\mathbf{N}$ строк $\mathbf{S_i}$. Тогда *уникальность* набора строк вычисляется как сумма непохожести по всем парам строк. Иными словами, эта величина равна сумме $\mathbf{d}(\mathbf{S_i}, \mathbf{S_j})$ для всех $1 \le \mathbf{i}, \mathbf{j} \le \mathbf{N}$.

Казимир Казимирович увлекается программированием. Его приняли на работу в «Interplanetary Software, Inc.». Первое задание – написать программу, которая вычисляет уникальность заданного набора строк. Напишите программу, вычисляющую уникальность заданного набора строк, чтобы Казимир Казимирович мог убедиться, что его программа работает правильно.

Формат входных данных

Первая строка содержит целое число $\mathbf{N}$ ($1 \le \mathbf{N} \le 200\,000$) – количество строк в наборе.

В каждой из следующих $\mathbf{N}$ строк записано по одной строке $\mathbf{S_i}$, состоящей из строчных букв латинского алфавита – сами строки.

Гарантируется, что суммарная длина всех $\mathbf{S_i}$ не превосходит 10^6 символов.

Формат выходных данных

Выведите ответ на задачу – уникальность заданного набора строк.

Примеры

Стандартный ввод	Стандартный вывод	Комментарий
4 b aab baaa ba	44	Посчитаем непохожесть по всем парам строк (учитываем, что d(A, B) = d(B, A) и d(A, A) = 0 для любых строк A и B): • d(‘b’, ‘aab’) = 4 • d(‘b’, ‘baaa’) = 3 • d(‘b’, ‘ba’) = 1 • d(‘aab’, ‘baaa’) = 7 • d(‘aab’, ‘ba’) = 5 • d(‘baaa’, ‘ba’) = 2 Ответ равен 2·(4+3+1+7+5+2) = 2·22 = 44
3 a ab aaaaa	20	Заметим, что: • d(‘a’, ‘ab’) = 1 • d(‘a’, ‘aaaaa’) = 4 • d(‘ab’, ‘aaaaa’) = 5 То есть, ответ равен 2·(1+4+5) = 20

Подзадача 1 (40 баллов)

N ≤100.

В этой подзадаче 20 тестов (с 3 по 22). Баллы за подзадачу выставляются только при прохождении всех тестов подзадачи.

Подзадача 2 (20 баллов)

N ≤5000.

В этой подзадаче 10 тестов (с 23 по 32). Баллы за подзадачу выставляются только при прохождении всех тестов подзадачи и предыдущей подзадачи.

Подзадача 3 (40 баллов)

Нет дополнительных ограничений.

В этой подзадаче 20 тестов (с 33 по 52). Тесты оцениваются независимо, но только в случае прохождения всех тестов предыдущих подзадач.

Получение информации о результатах окончательной проверки

Сообщаются только баллы за каждую подзадачу в целом.

Задача 4. Домики во“Флатландии”

Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Страну «Флатландия» населяют плоские жители, которые живут в плоских домиках на координатной плоскости. Все домики имеют форму прямоугольников. Все стороны прямоугольников параллельны осям координат, все вершины имеют целочисленные положительные координаты. Стороны различных прямоугольников не пересекаются, но могут иметь общую сторону или общую часть стороны, например, соприкасаться вершинами.

Необходимо определить, сколько домиков видно полностью и частично, если смотреть на них из начала координат (точка с координатами `0,0`)

Отрезок `AB`, где точки `A` и `B` принадлежат одной стороне прямоугольника (домика), считается видимым из точки `O`, если внутренние точки треугольника `OAB` не имеют общих точек с каким-либо другим прямоугольником (домиком).

Сторона домика видна, если виден отрезок `AB`, где `A` и `B` – начало и конец данной стороны домика.

Домик виден полностью, если полностью видна какая-либо его сторона.

Домик виден частично, если виден отрезок `AB` на какой-либо стороне домика, точки `A` и `B` различны.

Требуется написать программу, которая по полученной информации установит:

1) Сколько домиков видно полностью?

2) Сколько домиков видно частично?

Формат входных данных

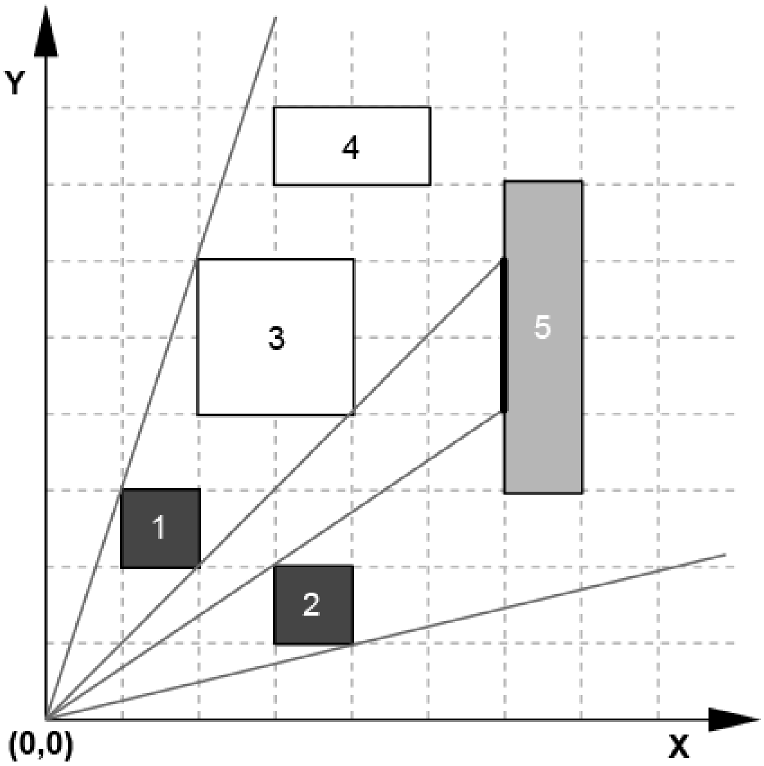
Первая строка содержит число P (1 или 2) — номер пункта задачи, ответ на который требуется получить. Вторая строка файла содержит число N ($1 \le N \le 1000$) – количество домиков. Каждая из последующих N строк содержит четыре числа $x1$, $y1$ $x2$, $y2$, разделенные пробелом, – координаты нижнего левого ($x1$, $y1$) и верхнего правого ($x2$, $y2$) угла домика в форме прямоугольника. Все координаты натуральные и не превышают 30000.

Формат выходных данных

Выведите количество видимых домиков (если P=1, то — количество полностью видимых домиков, если P=2, то — количество частично видимых домиков).

Примеры (*соответствуют рисунку ниже*)

Стандартный ввод	Стандартный вывод
1 5 1 2 2 3 3 1 4 2 2 4 4 6 3 7 5 8 6 3 7 7	2
2 5 1 2 2 3 3 1 4 2 2 4 4 6 3 7 5 8 6 3 7 7	3



Система оценки и описание подзадач

В этой задаче все тесты оцениваются независимо и баллы начисляются за каждый из них.

Получение информации о результатах окончательной проверки

Сообщается результат окончательной проверки на каждом тесте.