

## Задача Sochi. Реалити-шоу

Имя входного файла:	input.txt или стандартный поток ввода
Имя выходного файла:	output.txt или стандартный поток вывода
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Одно известное реалити-шоу набирает участников для третьего сезона! Собеседование прошли  $n$  кандидатов, пронумерованных от 1 до  $n$ .  $i$ -й кандидат обладает агрессивностью  $l_i$ , и для того, чтобы взять этого кандидата, организаторам шоу придется потратить  $s_i$  рублей.

Ведущая реалити-шоу листает досье всех кандидатов от  $i = 1$  до  $i = n$  в порядке возрастания номеров, и для каждого из них она принимает решение, стоит ли взять этого кандидата в шоу. Если агрессивность кандидата  $i$  строго выше, чем агрессивность какого-либо из уже **взятых** кандидатов, то кандидата  $i$  она точно брать откажется. В противном случае ведущая может на свой выбор как взять кандидата в шоу, так и не взять. Ведущая хочет выбрать участников таким образом, чтобы максимизировать *прибыль*.

Рассматриваемое реалити-шоу получает доход следующим образом. Для каждого значения агрессивности  $v$  указана доходность  $c_v$ , которая может быть как положительной, так и отрицательной. Участники, попавшие в проект, поочередно выходят на сцену в порядке возрастания номеров. Как только участник с номером  $i$  выходит на сцену, происходит следующее:

- Шоу зарабатывает  $c_i$  рублей, где  $l_i$  — изначальная агрессивность участника.
- Если оказывается, что на сцене находятся два участника с одинаковым значением агрессивности, то они немедленно начинают драку. В результате драке происходит следующее:
  - проигравшего участника увозит скорая помощь, и он выбывает из проекта,
  - агрессивность выигравшего участника возрастает на один, а шоу зарабатывает  $c_t$  рублей, где  $t$  — новое значение агрессивности.
- Драки на сцене продолжаются до тех пор, пока на сцене не останутся участники с различными значениями агрессивности.

Ведущая шоу хочет выбрать команду таким образом, чтобы максимизировать прибыль, которая определяется как суммарный доход от происходящего на сцене, минус суммарные затраты на приглашение участников (которые определяются как сумма  $s_i$  для выбранных участников). Помогите ведущей сделать шоу максимально прибыльным!

### Формат входных данных

В первой строке вводится два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 2000$ ) — количество кандидатов и максимальное начальное значение агрессивности кандидатов.

Во второй строке вводится  $n$  целых чисел  $l_i$  ( $1 \leq l_i \leq m$ ) — агрессивность каждого из кандидатов.

В третьей строке вводится  $n$  целых чисел  $s_i$  ( $0 \leq s_i \leq 5000$ ) — количество рублей, которое придется заплатить для найма каждого из кандидатов.

В четвертой строке вводится  $n + m$  целых чисел  $c_i$  ( $|c_i| \leq 5000$ ) — уровень доходности для каждого из значений агрессивности.

Гарантируется, что при данных ограничениях значение агрессивности участников не может превысить  $n + m$ .

### Формат выходных данных

Выведите одно целое число — максимальную прибыль шоу.

## Примеры

ВВОД	ВЫВОД
5 4 4 3 1 2 1 1 2 1 2 1 1 2 3 4 5 6 7 8 9	6
2 2 1 2 0 0 2 1 -100 -100	2
5 4 4 3 2 1 1 0 2 6 7 4 12 12 12 6 -3 -5 3 10 -4	62

## Пояснение

В первом примере выгоднее всего взять кандидатов с номерами 1, 2, 3, 5. В таком случае шоу заплатит им за участие  $1 + 2 + 1 + 1 = 5$  рублей. На сцене же будет происходить следующее:

- на сцене появится участник со значением агрессивности 4, шоу заработает 4 рубля;
- на сцене появится участник со значением агрессивности 3, шоу заработает 3 рубля;
- на сцене появится участник со значением агрессивности 1, шоу заработает 1 рубль;
- на сцене появится еще один участник со значением агрессивности 1, шоу заработает еще 1 рубль, начнется драка. Один из участников проиграет и покинет проект, а второй повысит свое значение агрессивности до 2. Шоу заработает на этом еще 2 рубля.

Таким образом, доход шоу составит  $4 + 3 + 1 + 1 + 2 = 11$  рублей, а суммарная прибыль будет равна  $11 - 5 = 6$  рублям.

Во втором примере мы не можем пригласить обоих кандидатов, так как значение агрессивности второго кандидата больше, поэтому нам выгоднее взять кандидата с номером 1.

## Система оценки

Тесты к этой задаче состоят из шести групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **необходимых** групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Дополнительные ограничения		Необх. группы	Комментарий
		$n$	$m$		
0	0	–	–	–	Тесты из условия.
1	14	$n \leq 15$	$m \leq 15$	0	
2	10	–	$m = 1$	–	
3	10	–	$m \leq 2$	2	
4	15	–	$m \leq 5$	2, 3	
5	26	$n \leq 200$	$m \leq 200$	0, 1	
6	25	–	–	0, 1, 2, 3, 4, 5	<b>Offline-проверка.</b>

## Задача Salt Lake City. Подарок

Имя входного файла: `input.txt` или стандартный поток ввода  
Имя выходного файла: `output.txt` или стандартный поток вывода  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 512 мегабайт

На восьмое марта Катерине подарили массив чисел. Со временем ей стало скучно просто смотреть на него, и она решила посчитать некоторые его бесполезные характеристики. Со всеми, что она придумывала, ей это удавалось. Придумав очередную — хог попарных сумм чисел в массиве, она поняла, что не может придумать, как посчитать её для большого массива, и попросила вас помочь. А вы сможете? Более формально, вам нужно посчитать

$$\begin{aligned} &(a_1 + a_2) \oplus (a_1 + a_3) \oplus \dots \oplus (a_1 + a_n) \oplus \\ &\quad \oplus (a_2 + a_3) \oplus \dots \oplus (a_2 + a_n) \oplus \\ &\quad \quad \quad \dots \\ &\quad \quad \quad \oplus (a_{n-1} + a_n) \end{aligned}$$

### Формат входных данных

Первая строка содержит целое число  $n$  ( $2 \leq n \leq 400\,000$ ) — количество чисел в массиве.

Вторая строка содержит целые числа  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^7$ ).

### Формат выходных данных

Выведите одно число — хог попарных сумм чисел в данном массиве.

### Примеры

ВВОД	ВЫВОД
2 1 2	3
3 1 2 3	2

### Пояснение

В первом примере есть всего одна сумма:  $1 + 2 = 3$ .

Во втором примере есть три суммы:  $1 + 2 = 3$ ,  $1 + 3 = 4$ ,  $2 + 3 = 5$ . В двоичной системе счисления это  $011_2 \oplus 100_2 \oplus 101_2 = 010_2$ , то есть 2.

$\oplus$  означает операцию побитового хог. Чтобы определить  $x \oplus y$  рассмотрим двоичную запись чисел  $x$  и  $y$ . Скажем, что  $i$ -й бит результата равен 1, если ровно один из  $i$ -х битов  $x$  и  $y$  равен 1. В противном случае  $i$ -й бит результата равен 0. Например,  $0101_2 \oplus 0011_2 = 0110_2$ .

### Система оценки

Тесты к этой задаче состоят из трёх групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **необходимых** групп.

Группа	Баллы	Дополнительные ограничения		Необх. группы	Комментарий
		$n$	$a_i$		
0	0	—	—	—	Тесты из условия.
1	34	$n \leq 1000$	—	0	
2	37	—	$1 \leq a_i \leq 100$	0	
3	29	—	—	0, 1, 2	

## Задача Монасо. Лапша быстрого приготовления

Имя входного файла:	<code>input.txt</code> или стандартный поток ввода
Имя выходного файла:	<code>output.txt</code> или стандартный поток вывода
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Ву проголодался после интенсивной тренировки и отправился в ближайший магазин, чтобы приобрести его любимую лапшу быстрого приготовления. После того, как Ву расплатился, кассир задал ему интересную задачу.

Дан двудольный граф, в вершинах **правой** доли которого записаны положительные целые числа. Для подмножества вершин  $S$  **левой** доли определим  $N(S)$  как множество вершин правой доли, смежной хотя бы с одной из вершин  $S$ , а  $f(S)$  — как сумму чисел, записанных в вершинах  $N(S)$ . Требуется найти наибольший общий делитель чисел  $f(S)$  для всех возможных непустых подмножеств  $S$ .

Ву слишком устал во время тренировки, чтобы справиться с такой задачей. Помогите ему!

### Формат входных данных

В первой строке записано целое число  $t$  ( $1 \leq t \leq 500\,000$ ) — количество тестовых случаев, для которых требуется решить задачу. Затем следует описание тестовых случаев.

В первой строке каждого описания записаны два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 500\,000$ ) — количество вершин в каждой из долей графа и количество рёбер, соответственно.

Во второй строке записано  $n$  целых чисел  $c_i$  ( $1 \leq c_i \leq 10^{12}$ ),  $i$ -е из которых задаёт число, записанное в  $i$ -й вершине правой доли графа.

В следующих  $m$  строках записаны пары целых чисел  $u_i$  и  $v_i$  ( $1 \leq u_i, v_i \leq n$ ), обозначающие ребро между  $u_i$ -й вершиной левой доли графа и  $v_i$ -й вершиной правой доли графа. Гарантируется, что в графе нет кратных рёбер.

Тестовые случаи разделены пустой строкой. Сумма значений  $n$  по всем тестовым случаям не превосходит 500 000, сумма значений  $m$  по всем тестовым случаям также не превосходит 500 000.

### Формат выходных данных

Для каждого тестового случая выведите единственное число — искомый наибольший общий делитель.

## Пример

ВВОД	ВЫВОД
3	2
2 4	1
1 1	12
1 1	
1 2	
2 1	
2 2	
3 4	
1 1 1	
1 1	
1 2	
2 2	
2 3	
4 7	
36 31 96 29	
1 2	
1 3	
1 4	
2 2	
2 4	
3 1	
4 3	

## Пояснение

Наибольшим общим делителем множества чисел называется наибольшее целое число  $g$  такое, что все элементы множества без остатка делятся на  $g$ .

В первом примере все вершины левой доли соединены ребром со всеми вершинами правой доли, поэтому значение  $f(S)$  для любого непустого подмножества будет равно 2, соответственно наибольший общий делитель этих значений будет также равен 2.

Во втором примере подмножество  $\{1\}$  вершин левой доли соединено ребром с вершинами  $\{1, 2\}$  правой доли, сумма значений в которых равна 2, а подмножество  $\{1, 2\}$  вершин левой доли соединено рёбрами с вершинами  $\{1, 2, 3\}$  правой доли, сумма значений в которых равна 3. Таким образом,  $f(\{1\}) = 2$ ,  $f(\{1, 2\}) = 3$ , что значит, что наибольший общий делитель всех чисел  $f(S)$  равен 1.

## Система оценки

Тесты к этой задаче состоят из трёх групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **предыдущих** групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Пусть  $\sum n$  обозначает сумму  $n$  по всем тестовым случаям, а  $\sum t$  сумму  $t$ .

Группа	Баллы	Дополнительные ограничения				Комментарий
		$n$	$t$	$\sum n$	$\sum t$	
0	0	–	–	–	–	Тесты из условия.
1	21	$n \leq 20$	$t \leq 400$	$\sum n \leq 100$	$\sum t \leq 2000$	
2	33	$n \leq 5000$	$t \leq 5000$	$\sum n \leq 10\,000$	$\sum t \leq 10\,000$	
3	46	–	–	–	–	<b>Offline-проверка.</b>

## Задача Rio de Janeiro. Латинский квадрат

Имя входного файла: `input.txt` или стандартный поток ввода  
Имя выходного файла: `output.txt` или стандартный поток вывода  
Ограничение по времени: 10 секунд  
Ограничение по памяти: 1024 мегабайта

Крис любит решать разные головоломки. Недавно он узнал про sudoku, идея которых основана на латинских квадратах. Таблица размера  $k \times k$  называется *латинским квадратом*, если количество различных элементов в таблице равно  $k$ , и ни в одной строке и ни в одном столбце нет двух одинаковых элементов.

Например, 

A	B
B	A

, 

S	P	R
R	S	P
P	R	S

 и 

Q
---

 являются латинскими квадратами, а 

A	B	C
C	C	A
C	A	B

,

A	B	C
D	C	A
C	A	B

 и 

A	B	C
C	A	B

 — нет.

Крис хочет сделать новую головоломку на основе латинских квадратов. Однако, у него есть только старая заготовка, которая является таблицей размера  $n \times m$ . Крис хочет вырезать из заготовки одну сплошную часть, которая будет являться латинским квадратом. Сколькими способами он может это сделать? Два способа считаются разными, если есть клетка таблицы-заготовки, которая принадлежит вырезаемой части в одном случае и не принадлежит в другом.

### Формат входных данных

В первой строке даны два целых числа  $n$  и  $m$  — размеры заготовки ( $1 \leq n, m \leq 2000$ ).

В следующих  $n$  строках даны строки  $s_i$  — описание заготовки. Каждая строка  $s_i$  состоит из  $2 \cdot m$  символов с ASCII-кодами от 33 до 126. В клетке заготовки на пересечении  $i$ -й строки и  $j$ -го столбца находится пара символов  $s_{i,2j-1}$  и  $s_{i,2j}$  ( $1 \leq i \leq n$ ,  $1 \leq j \leq m$ ). Элементы в двух клетках заготовки равны, если упорядоченные пары символов в этих клетках равны. За подробным пояснением входных данных смотрите замечания.

### Формат выходных данных

Выведите одно целое число — количество способов вырезать из заготовки латинский квадрат.

### Примеры

ВВОД	ВЫВОД
4 5 AABVAAAACC VVAABVSSAA AABVSSAABB VVSSAABVSS	26
5 10 !"#\$%&'()*+,-./01234 56789:;<=>?@ABCDEFGH IJKLMNOPQRSTUVWXYZ[\ ]^_`abcdefghijklmnop qrstuvwxyz{ }~!"#\$%&	50

### Пояснение

В первом примере помимо 20 способов вырезать латинский квадрат  $1 \times 1$ , существует еще 6 следующих способов вырезать латинский квадрат:

AA	BB	AA	AA	CC
BB	AA	BB	CC	AA
AA	BB	CC	AA	BB
BB	CC	AA	BB	CC

(a) 1-й способ

AA	BB	AA	AA	CC
BB	AA	BB	CC	AA
AA	BB	CC	AA	BB
BB	CC	AA	BB	CC

(b) 2-й способ

AA	BB	AA	AA	CC
BB	AA	BB	CC	AA
AA	BB	CC	AA	BB
BB	CC	AA	BB	CC

(c) 3-й способ

AA	BB	AA	AA	CC
BB	AA	BB	CC	AA
AA	BB	CC	AA	BB
BB	CC	AA	BB	CC

(d) 4-й способ

AA	BB	AA	AA	CC
BB	AA	BB	CC	AA
AA	BB	CC	AA	BB
BB	CC	AA	BB	CC

(e) 5-й способ

AA	BB	AA	AA	CC
BB	AA	BB	CC	AA
AA	BB	CC	AA	BB
BB	CC	AA	BB	CC

(f) 6-й способ

## Система оценки

Тесты к этой задаче состоят из пяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **необходимых** групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Доп. ограничения	Необх. группы	Комментарий
		$n, m$		
0	0	–	–	Тесты из условия.
1	9	$n, m \leq 20$	0	
2	10	$n, m \leq 100$	0, 1	
3	25	$n, m \leq 500$	0, 1, 2	
4	26	–	–	В каждой строке и каждом столбце все элементы различны.
5	30	–	0, 1, 2, 3, 4	<b>Offline-проверка.</b>