

Задача А. Совунья и соления

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

Крош и Ёжик собрали для Совуны n грибов. Для их засолки Совунья подготовила k банок и хочет разложить грибы поровну в каждую из них.

К сожалению, количество собранных грибов n может не делиться нацело на число банок k . Совунья не хочет выбрасывать грибы, поэтому просит Кроша и Ёжика принести ещё немного. Однако, чтобы не перетруждать друзей, она хочет попросить как можно меньше дополнительных грибов.

Формат входных данных

В первой строке входных данных вводится число $1 \leq n \leq 10^9$ — количество грибов, которые принесли Крош и Ёжик.

Во второй строке входных данных вводится число $1 \leq k \leq 10^9$ — количество банок, которые подготовила Совунья.

Формат выходных данных

Выведите минимальное количество грибов, которое Совунье надо попросить у своих друзей, чтобы можно было разложить грибы по банкам поровну.

Система оценки

В задаче 20 тестов, каждый из которых оценивается в 5 баллов.

Решения, корректно работающие для $1 \leq n, k \leq 10^5$, получают не менее 50 баллов.

Примеры

стандартный ввод	стандартный вывод
3 4	1
7 5	3

Замечание

В первом тесте Ёжик и Крош принесли 3 гриба. У Совуны 4 банки. Совунье надо попросить 1 гриб, тогда суммарно грибов станет 4, и она сможет в каждую банку положить 1 гриб.

Во втором тесте Крош и Ёжик принесли Совунье 7 грибов, у неё есть 5 банок. Ей необходимо попросить ещё 3 гриба, тогда всего грибов будет 10, и она сможет положить в каждую банку по 2 гриба.

Задача В. Числовые шестерёнки

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

Пин, гениальный изобретатель из Ромашковой долины, снова что-то мастерит. На этот раз он создал новый механизм для Биби и назвал его «Числовые шестерёнки». Сердце механизма — две пары шестерёнок, которые нужно соединить особым образом. Пин экспериментальным путём выяснил, что механизм запустится, если у двух соединённых пар совпадают их *гармонические числа*.

У Пина есть четыре шестерёнки, на каждой из которых написано одно число: на первой шестерёнке — a , на второй — b , на третьей — c и на четвёртой — d .

Гармоническое число упорядоченной пары шестерёнок (x, y) (где $y \neq 0$) — это остаток от деления x на y . Например, для пары $(7, 3)$ гармоническое число равно 1, так как $7 \bmod 3 = 1$.

Порядок в паре важен: для пары $(7, 3)$ гармоническое число равно 1, а для пары $(3, 7)$ — 3.

Помогите Пину понять, сможет ли он запустить механизм, используя эти четыре шестерёнки.

Формат входных данных

В четырёх строках вводятся числа a, b, c, d ($0 \leq a, b, c, d \leq 10^9$) — числа на шестерёнках Пина.

Формат выходных данных

Если возможно соединить шестерёнки, выведите две строки. В каждой строке — два числа (в нужном порядке), образующие пары с одинаковыми гармоническими числами.

Если запустить механизм невозможно, выведите -1 .

Если решений несколько, можно вывести любое.

Система оценки

Оценка складывается из баллов за пройденные тесты. За каждый правильно пройденный тест вы получаете 2 балла.

Примеры

стандартный ввод	стандартный вывод
7 5 3 9	9 7 5 3
2 3 2 2	-1

Замечание

В первом примере для первой пары гармоническое число $9 \bmod 7 = 2$.

Для второй пары — $5 \bmod 3 = 2$.

Два гармонических числа совпадают, значит, механизм можно запустить.

Задача С. Сонный Крош

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

Крош заботится о своём здоровье и записывает, сколько он спит. Каждый раз его умные часы фиксируют время засыпания и время пробуждения в формате ЧЧ:ММ. Обычно Крош ложится вечером, но иногда слишком долго читает книгу и засыпает уже после полуночи, а в выходные иногда даже спит днём.

Известно, что за один раз Крош спит меньше 24 часов, и время пробуждения всегда хронологически позже времени засыпания (возможно, на следующие сутки).

По записям за N дней определите, сколько всего часов и минут проспал Крош за это время.

Формат входных данных

В первой строке задано целое число N ($1 \leq N \leq 100$) — количество записей.

В каждой из следующих N строк содержатся два времени в формате ЧЧ:ММ, разделённые пробелом: время засыпания и время пробуждения. Гарантируется, что оба времени корректны ($00 \leq \text{ЧЧ} \leq 23$, $00 \leq \text{ММ} \leq 59$).

Формат выходных данных

Выведите два целых числа — общее количество часов и минут сна за все дни.

Система оценки

В задаче потестовая оценка: за каждый пройденный тест начисляется 5 баллов.

Пример

стандартный ввод	стандартный вывод
2 23:30 06:15 14:00 15:30	8 15

Замечание

В первом случае Крош спал с 23:30 до 06:15 следующего дня — это 6 часов 45 минут. Во втором — с 14:00 до 15:30, то есть полтора часа. Общее время сна: 8 часов и 15 минут.

Задача D. Глиняная табличка

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

Во время очередной экспедиции в пустыню Крош и Ёжик наткнулись на загадочную глиняную табличку, покрытую древними знаками. Поверхность таблички была аккуратно расчерчена на сетку из N строк и M столбцов. Некоторые клетки таблицы были закрашены. Друзья догадались, что каждый прямоугольник 5×3 может скрывать какую-то цифру от 0 до 9. Пример представления цифр (звёздочки обозначают закрашенные клетки, а точки — незакрашенные):

```
0: *** 1: *.* 2: *** 3: *** 4: *.*
   *.*   *.*   *.*   *.*   *.*
   *.*   *.*   ***   ***   ***
   *.*   *.*   *..   ..*   *.*
   ***   *.*   ***   ***   *.*

5: *** 6: *** 7: *** 8: *** 9: ***
   *..   *..   *.*   *.*   *.*
   ***   ***   *.*   ***   ***
   *.*   *.*   *.*   *.*   *.*
   ***   ***   *.*   ***   ***
```

Но, как это часто бывает с древними артефактами, всё оказалось не так просто! Похоже, авторы таблички (возможно, шумеры, а может, просто очень взволнованный кактус) в спешке закрасили лишние клетки.

Крош и Ёжик подумали и решили, что *прямоугольник 5×3 содержит цифру d , если все клетки, которые должны быть закрашены у цифры d , действительно закрашены на табличке. А если где-то добавились лишние звёздочки — ничего страшного! Главное, чтобы ничего не пропало.*

Например, прямоугольник:

```
***
.*.
***
**.
```

содержит цифру 2 — все её обязательные звёздочки на месте. А вот ни одна другая цифра не подходит — у неё обязательно найдётся хотя бы одна нужная клетка, которая осталась пустой.

А что, если прямоугольник подходит сразу под несколько цифр? Тогда выберем **самую большую!** — решили Крош и Ёжик. Например, прямоугольник:

```
***
***
***
.*.
***
```

подходит и под 1, и под 3, и под 5, и даже под 9. Но поскольку 9 — самая большая из них, именно 9 и будет записана в ответ.

Важно! Цифры не могут быть перевернуты и всегда расположены вертикально.

Теперь друзья хотят узнать: **чему равна сумма всех цифр**, закодированных в табличке?

Для этого нужно рассмотреть все возможные позиции прямоугольника 5×3 , которые полностью помещаются в таблицу — двигая его по строкам сверху вниз и по столбцам слева направо, как будто

«скользя» по поверхности. Прямоугольники могут пересекаться — это не разбиение на блоки, а именно все возможные окна размером 5×3 .

Если в каком-то окне не получается распознать ни одну цифру, оно просто ничего не добавляет к сумме.

Помогите Крошу и Ёжику расшифровать древнее число!

Формат входных данных

В первой строке вводится два числа $5 \leq N \leq 100$ и $3 \leq M \leq 100$.

В следующих N строках вводятся строки длины M , состоящие из символов звездочек (*) и точек (.) — исходная таблица.

Формат выходных данных

Выведите одно число — сумму всех цифр, закодированных в таблице.

Система оценки

В задаче 50 тестов, каждый оценивается на 2 балла. Тесты разбиты на 3 группы, внутри каждой группы действует потестовая оценка, то есть баллы начисляются за каждый тест независимо. Группа 3 тестируется, только если все тесты первой и второй группы были пройдены.

Группа	Ограничения	Доп. условия	Баллы	Зависимые группы
0	Тесты из условия	-	0	
1	$5 \leq N \leq 100, 3 \leq M \leq 100$	Только цифра 1*	20	
2	$N = 5, M = 3$	-	40	
3	$5 \leq N \leq 100, 3 \leq M \leq 100$	-	40	0, 1, 2

* Для каждого прямоугольника 5×3 верно, что либо в нём нет цифры, либо закодирована цифра 1.

Примеры

стандартный ввод	стандартный вывод
5 3 *** *.. *** ..* ***	5
6 6 .***.* .*...* .***.* .*.*.* .***.***	8
10 10 ..*..... ..*..... ..*..... ..*.*.*. ..*.*.*. ..*.*.*. ..*.*.*. ..*.*.*. ..*.*.*. ..*..... ..*.....	11

Замечание

В первом примере явно закодирована цифра 5.

Во втором примере входных данных в прямоугольнике, с левым верхним углом в клетке с координатами (1, 2) закодирована цифра 6. И в прямоугольниках с верхними левыми углами в координатах (1, 4) и (2, 4) закодирована цифра 1. Сумма всех закодированных чисел равна 8.

Задача Е. Модница

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

Нюша — большая модница и поклонница косметики. В её коллекции сотни баночек с пудрами, помадами и кремами. Перед каждой прогулкой с Барашем она наносит макияж, чтобы подчеркнуть свою элегантность.

Как известно любой моднице, не все косметические средства хорошо сочетаются друг с другом. Чтобы навести порядок в своей огромной коллекции, Нюша расставила всю косметику в один длинный ряд. Каждой баночке i она сопоставила целое число a_i — числовую характеристику её стиля.

Нюша придумала особое правило для оценки сочетаемости средств: для любого набора баночек она вычисляет так называемое **магическое число** — результат операции **XOR** ($\oplus$, побитовое исключающее ИЛИ) над всеми числами в этом наборе.

Что такое XOR?

XOR — это особая операция над числами. Чтобы её выполнить, нужно:

- записать числа в двоичной системе (с одинаковым числом цифр, добавляя нули слева при необходимости);
- в каждом разряде сравнить цифры:
 - если цифры одинаковые (0 и 0 или 1 и 1) — пишем 0;
 - если разные (0 и 1) — пишем 1.

Например:

$$\begin{array}{rcl} 5 & = & 101_2 \\ 3 & = & 011_2 \\ \hline 5 \oplus 3 = 6 & = & 110_2 \end{array}$$

Для трёх и более чисел XOR вычисляется по шагам: сначала XOR первых двух, потом XOR результата с третьим и так далее.

В большинстве языков программирования операция XOR записывается с помощью символа «крышка». Например, в Python, C++ или Java выражение $a \wedge b$ означает побитовое исключающее ИЛИ чисел a и b .

Поскольку ряд косметики получился очень длинным, для одного макияжа Нюша решила использовать только средства, стоящие подряд, то есть некоторый подотрезок в этом ряду. Прогулок с Барашем предстоит много, и Нюша хочет каждый раз выглядеть по-новому.

Нюша хочет рассмотреть **все возможные** подотрезки (все способы выбрать подряд идущие баночки), вычислить магическое число для каждого из них, а затем найти XOR **всех этих магических чисел**. Помогите Нюше найти это итоговое число!

Формат входных данных

В первой строке находится одно целое число n ($1 \leq n \leq 2 \cdot 10^5$) — количество косметических средств.

Во второй строке находятся n целых чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^9$) — числовые характеристики стиля средств.

Формат выходных данных

Выведите одно целое число — XOR магических чисел всех подотрезков.

Система оценки

В задаче 50 тестов, каждый оценивается на 2 балла. Тесты разбиты на 3 группы, внутри каждой группы действует потестовая оценка, то есть баллы начисляются за каждый тест независимо. Зависимости между группами тестов указаны в таблице ниже

Группа	Ограничения	Баллы	Зависимые группы
0	Тесты из условия	0	
1	$1 \leq n \leq 100$	30	0
2	$1 \leq n \leq 5000$	30	0, 1
3	$1 \leq n \leq 2 \cdot 10^5$	40	0, 1, 2

Примеры

стандартный ввод	стандартный вывод
3 1 2 3	2
5 1 3 4 5 2	7
9 13 15 134 20 11 142 64 32 10	202

Замечание

Разберём первый пример: массив $[1, 2, 3]$.

Все подотрезки и их магические числа:

- $[1]$: 1
- $[2]$: 2
- $[3]$: 3
- $[1, 2]$: $1 \oplus 2 = 3$
- $[2, 3]$: $2 \oplus 3 = 1$
- $[1, 2, 3]$: $1 \oplus 2 \oplus 3 = 0$

Теперь вычислим XOR всех этих значений:

$$1 \oplus 2 \oplus 3 \oplus 3 \oplus 1 \oplus 0 = 2.$$

Ответ: 2.