

Задача 1 Football (4 варианта)

Пытаясь вспомнить первую шестёрку чемпионата района по футболу 2019 года, пятеро болельщиков сказали, что, по их мнению:

Рассвет был вторым, Урожай пятыми.
Шайба была второй, Метеор третьим.
Рассвет был третьим, Зубило шестым.
Пишевик был первым, Урожай третьими.
Шайба была третьей, Зубило четвёртым.

При просмотре сайта Википедии выяснилось, что каждый из болельщиков, ошибся в одном из высказываний, а в другом был прав. Каково было истинное распределение мест в чемпионате, если никакие две команды не делили одно место.

Задача 2 Sophie Germain

Простое число Софи Жермен – такое простое число p , что число $2p + 1$ также простое. Найдите все простые числа Софи Жермен в заданном отрезке.

Входные данные:

Первая и единственная строка содержит два натуральных числа A , B ($A, B < 10^6$) записанных через пробел

Выходные данные:

Выведите в единственной строке все простые числа Софи Жермен в интервале от A до B (A и B входят в интервал).

Пример:

Входные данные:

23 100

Выходные данные:

23 29 41 53 83 89

Примечание: <https://oeis.org/A005384>

2, 3, 5, 11, 23, 29, 41, 53, 83, 89, 113, 131, 173, 179, 191, 233, 239, 251, 281, 293, 359, 419, 431, 443, 491, 509, 593, 641, 653, 659, 683, 719, 743, 761, 809, 911, 953, 1013, 1019, 1031, 1049, 1103, 1223, 1229, 1289, 1409, 1439, 1451, 1481, 1499, 1511, 1559

Задача 3. Спутник "Заря"

На спутнике «Заря» установлен прибор, предназначенный для измерения количества метеоритов в заданном секторе. Каждую минуту прибор передаёт по каналу связи натуральное число – количество метеоритов, полученных в заданном секторе за

последнюю минуту. Временем, в течение которого происходит передача, можно пренебречь. Необходимо найти в заданной серии количество пар таких показаний прибора, произведение которых кратно 6 и между моментами передачи которых прошло не менее T минут. Количество метеоритов, которое считает прибор за минуту, не превышает 2^{64} . Общее количество показаний прибора в серии не превышает 10^6 .

Напишите программу для решения поставленной задачи, которая будет эффективна как по времени, так и по памяти (или хотя бы по одной из этих характеристик). Входные данные представлены следующим образом. В первой строке задаётся число N ($N \leq 10^6$) – общее количество показаний прибора и T ($T \leq 1000$) – количество минут. Гарантируется, что $N > T$. В каждой из следующих N строк задаётся одно натуральное число – очередное показание прибора.

Пример входных данных:

```
5 3
12
8
10
7
9
```

Пример выходных данных для приведённого выше примера входных данных:

```
3
```

Задача 4. Вероятностные драконы

Великие конструкторы Трурль и Клапауциус, изучая вероятностных драконов, убедились, что драконов, конечно, не бывает, но некоторых из них не бывает гораздо сильнее, чем других.

В ходе опытов и наблюдений Трурль выявил некоторое количество свойств, которыми вероятностный дракон может обладать, а может и не обладать, и предложил обозначать каждое такое свойство латинской буквой. Клапауциус в свою очередь выделил такие комбинации драконьих свойств или отсутствия таковых, при которых такого дракона не просто не бывает (как и любых других), но, более того, не бывает настолько сильно, что никакие воздействия на вероятности не могут ничего с этим поделать. Например, если дракон имеет шесть лап, не имеет крыльев и обладает шкурой белого цвета, то, хотя его и не бывает, но живёт он при этом, скорее всего, на астероидах в безвоздушном

пространстве -- а если и это окажется не так, то такого дракона не бывает вообще совсем никогда.

В дальнейших исследованиях комбинации драконьих свойств стали обозначать словами, состоящими из латинских букв, причём строчная буква соответствовала наличию свойства, а заглавная -- его отсутствию. Так, если обозначить шестилапость буквой *h*, наличие крыльев буквой *w*, белый цвет шкуры буквой *p*, а манеру жить на безвоздушных астероидах буквой *a*, то получится, что дракона, соответствующего слову *hWpA*, не бывает намного сильнее, нежели прочих.

Наработав базу данных по таким "совсем запрещённым" комбинациям, исследователи озаботились вопросом, а сколько же ещё разнообразных (отличающихся друг от друга хотя бы одним свойством) драконов не бывает, но при этом не так чтобы очень.

На вход вашей программе будет подана последовательность слов, возможно, имеющих разную длину, и при этом разделённых пробельными символами -- пробелами и переводами строки. Пробельных символов между двумя любыми словами может быть сколько угодно. Буквы в словах могут стоять в любом порядке, это ни на что не влияет. Ввод заканчивается ситуацией "конец файла"; может быть задействовано любое количество латинских букв (драконьих свойств) от одной до 26 (весь алфавит).

Ваша программа должна напечатать целое число, обозначающее, сколько разнообразных комбинаций из тех свойств, какие были упомянуты во входной последовательности, можно построить так, чтобы соответствующих драконов хотя и не бывало, но не так чтобы прямо совсем.

Например:

Ввод: abC

вывод: 7

Ввод: mXt TMx

вывод: 6

Ввод: aB CD

вывод: 9

Ввод: aB CD ad BCd

вывод: 6

Задача 5. Virus

Как известно, наследственная информация во многих вирусах хранится в молекуле РНК. РНК представляет собой молекулу-полимер, то есть цепочку, составленную из четырех оснований: аденина (A), гуанина (G), урацила (U) и цитозина (C). Таким образом, РНК можно записать как строку, составленную из букв A, G, U, C, например, AGGAU.

Будем считать, что длина РНК – M оснований, причем РНК разбивается на гены равной длины L оснований, то есть РНК состоит из M/L генов. Вирус содержит одну РНК.

Обнаружилось, что некоторый вирус способен к размножению следующим образом: из двух вирусов, то есть из двух РНК образуется третья РНК нового вируса, которая является случайной комбинацией генов двух первых вирусов, расположенных на соответствующих местах.

Например, при $M = 6$ и $L = 2$ и двух РНК AGUUCA и CUGAGC могут образоваться РНК AGGAGC или CUUUCA, **но не** могут образоваться AUGAGC (можно обмениваться только L основаниями) или UUGCCU (гены нельзя переставлять).

Из тканей организма-хоста были выделены N РНК вирусов ($4 < N \leq 100$), причем известно, что все РНК, за исключением не более чем одной РНК были получены описанным выше способом из некоторой пары РНК.

Определите номер той РНК во входном наборе, которая не может быть получена способом, описанным выше.

Входные данные задаются на стандартном потоке ввода следующим образом: в первой строке задаются числа M ($M \leq 100000$), L и N . Затем идут M строк текста, и в каждой строке текста задаются основания РНК, находящиеся на соответствующем месте в РНК. Строка не содержит пробелов. Например,

6 2 5

ACAAC

GUUGU

UGGGU

UAAAU

CGGGC

ACCCA

Обратите внимание, что цепочки, составляющие РНК, записаны "по вертикали".

На стандартный поток вывода напечатайте номер "чужеродной" РНК (РНК нумеруются от 1), или 0, если такой РНК в наборе нет. Гарантируется, что, если чужеродная РНК в наборе есть, она определяется однозначно.

Examples

Input

6 2 5

ACAAC

GUUGU

UGGGU

UAAAU

CGGGC

ACCCA

Output

3

Input

6 2 5

ACAAC

GUGGU

Олимпиада "Ломоносов" по информатике
Заключительный этап 7-10. Москва
12 мая 2020

UGUGU
UAUUAU
CGGGC
ACCCA
Output
0