

Задача А. Муравьиный десант

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

Полный муравьев вертолет летит под ядерным дождем над кольцевой дорогой.

Дорога состоит из N клеток, пронумерованных от 1 до N по часовой стрелке. Изначально вертолет находится над клеткой 1, и из него в эту клетку десантируется первый муравей. Он начинает бежать по дороге по часовой стрелке, перемещаясь за одну минуту ровно на одну клетку. Через K минут муравей погибает, а в клетке, в которой он погиб, сразу же вырастает гриб. После гибели очередного муравья вертолет перемещается на следующую клетку по часовой стрелке (относительно своего положения) и высаживает в неё следующего муравья, который бежит аналогичным образом.

По дороге муравей съедает все грибы. При этом первый съеденный гриб позволяет ему пробежать дополнительно $\lfloor K/2 \rfloor$ (округление вниз) минут, второй — дополнительно $\lfloor K/3 \rfloor$ минут и т.д. Другими словами, если муравей съел всего P грибов, то он пробежит $K + \lfloor K/2 \rfloor + \lfloor K/3 \rfloor + \dots + \lfloor K/(P+1) \rfloor$ минут. Если муравей собирается погибнуть в клетке с грибом, он успевает его съесть. Таким образом, ни в какой клетке не может оказаться более одного гриба в один момент времени.

Если очередной муравей пробегает всю дорогу и возвращается в свою стартовую клетку, миссия считается выполненной, муравью присваивается звание героя и вертолет улетает. Необходимо определить номер муравья-героя или вывести -1 , если вертолет будет кружить над дорогой вечно, при условии, что в вертолете изначально бесконечное количество муравьев.

Формат входных данных

В единственной строке заданы два целых числа: количество клеток N ($2 \leq N \leq 10^9$) и число K ($1 \leq K \leq N$).

Формат выходных данных

Порядковый номер муравья-героя (нумерация с единицы).

Примеры

стандартный ввод	стандартный вывод
7 4	4
5 3	-1

Замечание

Рассмотрим первый тестовый пример.

Первый муравей начнёт свой путь в клетке 1, а погибнет в клетке под номером 5. Второй муравей начнёт путь в клетке 2, съест гриб в клетке 5 и погибнет в клетке 1. Третий муравей начнёт путь в клетке 3, погибнет в клетке 7. Четвёртый муравей начнёт путь в клетке 4, съест грибы в клетках 7 и 1, а закончит путь в клетке 4, получив звание героя.

Задача В. Фиксированная цена

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

В городе открылся магазин с фиксированной ценой P , в котором действуют следующие правила:

- если рыночная стоимость единицы товара S_i больше либо равна P , то одна единица товара продаётся по цене P ;
- если рыночная стоимость единицы товара S_i меньше P , то товар продаётся только целыми упаковками ценой P , каждая из которых содержит $\lceil P/S_i \rceil$ (округление вверх) единиц товара.

Посчитайте для каждого из N товаров стоимость покупки не менее A_i единиц товара в магазине с фиксированной ценой.

Формат входных данных

В первой строке задано количество товаров N ($1 \leq N \leq 1000$) и фиксированная цена P ($1 \leq P \leq 1000$).

Далее следуют N строк, содержащих описания товаров: требуемое количество товара A_i и рыночную стоимость единицы товара S_i ($1 \leq A_i, S_i \leq 1000$).

Все числа во входных данных целые.

Формат выходных данных

Выведите N чисел — стоимость покупки не менее A_i единиц товара в магазине с фиксированной ценой.

Пример

стандартный ввод	стандартный вывод
5 100	100
1 101	300
6 51	200
11 10	100
12 9	400
4 100	

Задача С. Как перестать беспокоиться и полюбить кактусы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Берляндская компания «Панические технологии» производит определенную стратегически важную продукцию. Единственный завод компании находится в городе 1, из которого продукция доставляется во все остальные города на Белых поездах по железнодорожным путям.

Всего в Берляндии N городов, некоторые из которых соединены ж/д путями. Города (вершины) и ж/д пути (ребра) образуют граф, который по необъяснимому стечению обстоятельств является *вершинным кактусом* — связным неориентированным графом, в котором каждая вершина лежит не более чем на одном простом цикле. Кроме того, длина всех циклов в этом графе **четная**.

Каждый поезд движется из города, в котором находится завод, в город назначения по такому маршруту, который позволит посетить минимально возможное количество городов, а если существует несколько таких маршрутов, выбирается случайно один из них. Когда очередной поезд прибывает в пункт назначения, его груз выгружают в бункер, а сам поезд разбирается на месте и больше не используется. Изначально такой алгоритм движения гарантирует, что по каждому ж/д пути поезда будут двигаться в одном направлении, что является необходимым и достаточным условием отсутствия возможности столкновения поездов.

Компания планирует построить еще K заводов в некоторых городах, из которых продукция будет так же доставляться во все остальные города по тому же алгоритму. Чтобы сохранить гарантию отсутствия возможности столкновения между поездами, компании может потребоваться достроить дополнительные ж/д пути. По законам Берляндии между городами A и B разрешено строить новый ж/д путь только в том случае, если непосредственно между ними уже существует ж/д путь. Таким образом, компания решила достроить между некоторыми такими городами парный ж/д путь, предписывая поездам двигаться из A в B по одному из путей, а из B в A — по другому. В таком случае столкновения между этими городами никогда не произойдет.

Необходимо определить, какое минимальное количество дополнительных законных ж/д путей должна построить компания, чтобы запустить очередной завод и гарантировать, что между поездами не может произойти столкновений.

Формат входных данных

В первой строке заданы числа N и M — количество городов Берляндии и количество существующих ж/д путей ($2 \leq N \leq 10^5$, $1 \leq M \leq N - 1 + N/4$).

Следующие M строк содержат по паре чисел A_i и B_i — номера городов, между которыми проложен ж/д путь ($1 \leq A_i, B_i \leq N$).

Гарантируется, что граф является вершинным кактусом, не содержит петель и кратных ребер, а все циклы в графе имеют четную длину.

Следующая строка содержит число K — количество заводов, которое компания планирует построить ($1 \leq K \leq 10^5$).

В последней строке перечислены номера городов X_j , в которых будут построены заводы в хронологическом порядке ($1 \leq X_j \leq N$).

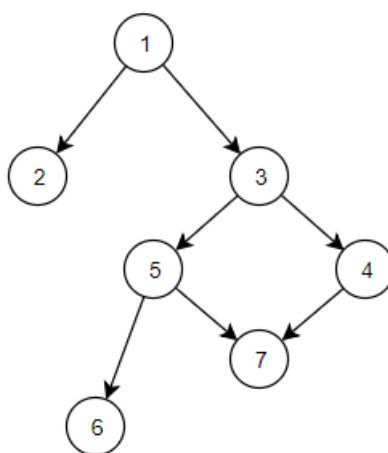
Формат выходных данных

Выведите K чисел Y_j — минимальное количество дополнительных ж/д путей, которые необходимо построить для запуска завода номер j .

Пример

стандартный ввод	стандартный вывод
7 7 1 2 1 3 3 4 4 7 5 7 3 5 5 6 5 6 2 4 4 5	4 1 2 0 0

Замечание



На рисунке обозначены города и начальное расположение путей и направлений, по которым проходят поезда из первого примера.

После постройки завода в городе 6, необходимо добавить пути между парами городов: $\langle 6, 5 \rangle$, $\langle 5, 3 \rangle$, $\langle 7, 4 \rangle$, $\langle 3, 1 \rangle$.

После постройки завода в городе 2, необходимо добавить пути между парой городов $\langle 2, 1 \rangle$.

После постройки первого завода в городе 4, необходимо добавить пути между парами городов: $\langle 4, 3 \rangle$, $\langle 7, 5 \rangle$.

Задача D. Евровидение

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Организаторы музыкального конкурса Евровидение устали от экспериментов участников и опубликовали новый свод правил, которому необходимо следовать при сочинении песен:

- длительность песни должна составлять ровно T секунд;
- куплеты, припев и проигрыши должны исполняться целиком без пауз;
- припев может быть исполнен сколько угодно раз (в том числе ноль), но каждый куплет и проигрыш — не более одного раза;
- куплеты, припев и проигрыши можно исполнять в любом порядке, но никакие два куплета не должны исполняться подряд.

Группа уже сочинила припев длительностью A и N проигрышей с длительностями S_i . Теперь группа планирует сочинить некоторое количество куплетов длительностью B , чтобы составить песню, удовлетворяющую всем правилам Евровидения. Какое минимальное количество куплетов ей необходимо сочинить?

Формат входных данных

В первой строке заданы три числа: требуемая длительность песни T ($1 \leq T \leq 10^{18}$), длительность припева A и длительность куплетов B ($1 \leq A, B \leq 500$).

Во второй задано количество проигрышей N ($0 \leq N \leq 500$).

Если $N > 0$, то в последней строке заданы N чисел S_i ($1 \leq S_i \leq 500$) — длительности сочинённых проигрышей.

Все числа во входных данных целые.

Формат выходных данных

Минимальное количество куплетов, которые необходимо сочинить группе.

Если ни при каком количестве куплетов невозможно сочинить песню, удовлетворяющую всем правилам, выведите -1.

Примеры

стандартный ввод	стандартный вывод
100 11 20 3 13 7 24	3
10 5 1 3 2 5 3	0
8 9 2 2 1 2	-1
10 3 10 0	1

Замечание

В первом тесте можно составить песню из следующих частей: припев, куплет, припев, куплет, припев, куплет, второй проигрыш.

Во втором тесте возможны следующие решения — дважды припев; все проигрыши; припев и второй проигрыш и т.д.

Задача Е. Скрщня

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	512 мегабайт

Программист Пётр ненавидит сокращения. Его предшественник, напротив, считал, что код должен быть коротким, и активно пользовался сокращениями. К счастью, перед увольнением он составил **словарь** — список S , состоящий из N полных слов, которые он сокращал.

У Петра есть текст T , состоящий из M слов. Слово из текста T_i является сокращением слова из словаря S_j , если выполняются следующие условия:

- T_i является подпоследовательностью S_j , то есть слово T_i можно получить путём удаления нуля или более символов из слова S_j ;
- не существует другого слова в словаре S_k ($k \neq j$), для которого слово T_i являлось бы подпоследовательностью.

Помогите Петру избавиться от сокращений, заменив каждое из них соответствующим полным словом из словаря. Если слово в тексте не является сокращением, оставьте его без изменений.

Формат входных данных

В первой строке задано количество слов в словаре N ($1 \leq N \leq 500$).

В следующих N строках заданы слова из словаря S , по одному в строке. Сумма длин всех слов в словаре не превосходит 2×10^6 .

В следующей строке задано количество слов в тексте M ($1 \leq M \leq 2000$).

В следующих M строках заданы слова из текста T , по одному в строке. Длина каждого слова в тексте не превосходит 10 символов ($1 \leq |T_i| \leq 10$).

Все слова в словаре и тексте состоят только из строчных букв латинского алфавита.

Формат выходных данных

Выведите M строк — слова текста после замены всех сокращений на полные слова.

Гарантируется, что сумма длин всех слов в тексте после замены сокращений не превысит 2×10^6 .

Пример

стандартный ввод	стандартный вывод
4	strtoint
abc	aa
strtoint	ababa
aba	ababa
ababa	abc
5	
sti	
aa	
aaa	
bb	
abc	

Задача F. Тайм-лимит

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

При подготовке задачи по программированию было разработано N решений и измерено время работы T_i каждого из них на самом худшем тесте.

Специальная строка S длины N описывает тип каждого решения: символ S_i определяет тип решения номер i . При этом символ 0 соответствует плохому решению, ? — спорному, а 1 — хорошему.

Тайм-лимит X считается корректным, если:

- время работы всех плохих решений превышает тайм-лимит как минимум в два раза ($T_i \geq X \times 2$, если $S_i = 0$);
- время работы всех хороших решений не превышает половину тайм-лимита ($T_i \leq X/2$, если $S_i = 1$).

Время работы спорных решений не влияет на корректность тайм-лимита.

Даны Q запросов, каждый из которых представляет собой строку S длиной N , описывающую тип каждого решения. Найдите для каждого запроса минимальный целый положительный тайм-лимит, который считается корректным, или определите, что это невозможно.

Формат входных данных

В первой строке задано количество решений N ($1 \leq N \leq 1000$).

Во второй строке заданы N чисел T_i ($1 \leq T_i \leq 10^6$) — время работы каждого из решений.

В третьей строке задано количество запросов Q ($1 \leq Q \leq 1000$).

В следующих Q строках содержатся запросы. Каждый запрос представляет собой строку S ($|S| = N$), которая содержит только символы 0, 1 или ? — описание типа каждого решения.

Все числа во входных данных целые.

Формат выходных данных

Выведите Q строк с ответами на запросы, согласно условию.

Если корректного тайм-лимита для запроса не существует, выведите для него -1.

Пример

стандартный ввод	стандартный вывод
5	200
500 1000 300 700 100	-1
6	1000
?0?01	2000
?0101	1
1?1?1	1
11111	
00000	
?????	

Задача G. Гирлянда

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

Новогодняя гирлянда после включения загорается на A минут, потом гаснет на A минут, потом опять загорается на A минут, гаснет на A , и так до бесконечности. Параметр A задаётся пользователем перед первым включением гирлянды и может быть любым натуральным числом.

Детишки расстроятся, если во время праздников, которые начинаются в момент времени 0 и заканчиваются в момент времени T , гирлянда будет гореть меньше половины времени.

В гости к детишкам на праздники приезжает дедушка, который не любит горящие гирлянды. Вам известны интервалы времени, в которые он будет присутствовать дома. Найдите такое значение параметра A и время включения гирлянды, которые позволят минимизировать длительность времени нахождения дедушки дома с включенной гирляндой и при этом не приведут к расстройству детишек. Учтите, что включить гирлянду можно в любой целочисленный момент времени, в том числе до наступления праздников, но выключить её после включения уже нельзя.

Формат входных данных

Первая строка содержит длительность новогодних праздников T ($1 \leq T \leq 5000$) в минутах и количество интервалов N ($0 \leq N \leq T/2$).

В следующих N строках заданы интервалы времени, в которые дедушка будет присутствовать дома, — начало интервала L_i и конец интервала R_i ($0 \leq L_i < R_i \leq T$). Интервалы не пересекаются и упорядочены по возрастанию времени начала ($R_i < L_{i+1}$).

Все числа во входных данных целые.

Формат выходных данных

Выведите три числа: минимальную длительность времени, которую дедушка вынужден будет провести дома с включённой гирляндой, искомое значение параметра A и момент времени включения гирлянды.

Если существует несколько решений, выведите решение с минимальным значением A . Если и таких решений существует несколько, выведите решение с самым поздним временем включения гирлянды.

Примеры

стандартный ввод	стандартный вывод
10 2 1 4 7 10	2 1 0
8 2 1 3 5 7	0 2 -1
6 1 0 4	1 3 3
5 1 0 5	3 1 0
4 0	0 1 1

Задача Н. LOCALC++

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

Ученые Берляндии разработали новый передовой защищенный отечественный язык программирования LOCALC++. В целом, этот язык является клоном языка C++ с той лишь разницей, что в LOCALC++ числа выводятся в консоль с разделителями.

В данной задаче рассматриваются только неотрицательные целые числа. При выводе число разбивается на группы из трех цифр, начиная с младших разрядов, и каждая группа отделяется пробелом. Например, число 178489 будет выведено в виде 178 489, число 17009 в виде 17 009, а число 5 будет выведено в таком же виде.

Программа управления атомными электростанциями Берляндии, написанная на LOCALC++, вывела в лог очень важную статистику в виде набора чисел через пробел. Известно, что исходные числа были строго меньше 10^K . Необходимо определить количество различных наборов чисел, вывод которых бы привел к такому же логу.

Формат входных данных

В первой строке задано число N , определяющее количество входных групп цифр и число K ($1 \leq N \leq 2 \cdot 10^5$, $3 \leq K \leq 6 \cdot 10^5$).

Во второй строке задано N групп цифр через пробел — лог программы управления атомными электростанциями Берляндии.

Формат выходных данных

Необходимо вывести количество возможных исходных наборов чисел с учетом того, что они могли быть только строго меньше 10^K . Гарантируется, что хотя бы один такой набор существует. Так как результат может быть достаточно большим, его необходимо вывести по модулю $10^9 + 7$.

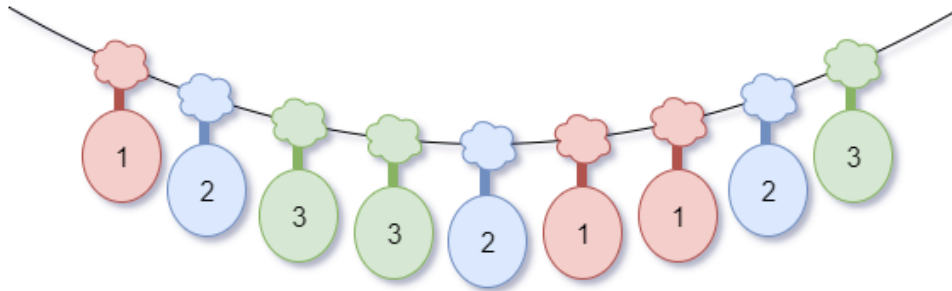
Примеры

стандартный ввод	стандартный вывод
8 7 10 500 303 4 507 89 654 003	6
3 6 328 032 0	1

Задача I. Хаотичные плюмбусы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Пока Рик в очередной раз спасал вселенную исключительно в своих интересах, Морти нашел в гараже набор разноцветных плюмбусов, развешенных на веревке.



Морти знает, что если плюмбусы разных цветов висят рядом, то они быстрее портятся, поэтому он решил минимизировать ущерб, то есть перевесить плюмбусы так, чтобы они были сгруппированы по цветам.

Всего на веревке $N \times K$ плюмбусов: по N штук каждого из K цветов. За одну секунду Морти может снять любой плюмбус и повесить его с левого или с правого края веревки, то есть левее или правее всех остальных. Морти хочет получить такое расположение плюмбусов, при котором все плюмбусы одного цвета формируют ровно одну последовательную группу. В итоге на веревке будет K таких групп. Последовательность групп при этом не имеет значения.

Определите, какое минимальное количество времени потребуется Морти, чтобы достичь своей цели.

Формат входных данных

В первой строке задано два целых числа N и K — количество плюмбусов каждого цвета и количество цветов соответственно ($1 \leq N, K \leq 1000$).

Во второй строке задано $N \times K$ чисел — начальное расположение плюмбусов, в котором каждое число A_i обозначает цвет плюмбуса номер i ($1 \leq A_i \leq K$).

В связи с большим объемом входных данных рекомендуется использовать эффективные методы ввода (например, `scanf` в C++).

Формат выходных данных

Необходимо вывести одно число — минимальное количество секунд, которое потребуется Морти для группировки плюмбусов описанным образом.

Примеры

стандартный ввод	стандартный вывод
3 3 1 2 3 3 2 1 1 2 3	4
2 4 3 3 1 1 4 4 2 2	0

Замечание

На рисунке изображен первый пример. Оптимальный алгоритм группировки для этого примера может выглядеть так: перевесить два плюмбуса цвета 1 из середины влево и два плюмбуса цвета 3 из середины вправо.

Задача J. Кормление крокодилов

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

В сафари-парке Вася увидел двух крокодилов, которых кормили туристы. Заметив, что один из крокодилов сильнее другого и съедает больше мяса, он решил восстановить справедливость.

Для того чтобы покормить крокодилов, Вася купил цельный кусок мяса весом N килограммов. Он решил разрезать его на равное количество кусков размерами A и B килограммов, причём так, чтобы ничего лишнего не осталось. Вася планирует каждые K секунд бросать крокодилам один кусок мяса, чередуя размеры кусков. Самым первым он бросит кусок размером A .

Крокодилы едят мясо по следующим правилам:

- если крокодил не занят поеданием схваченного ранее куска мяса, и бросается новый кусок, то он пытается его схватить;
- если только один крокодил пытается схватить кусок, то он достаётся ему;
- если оба крокодила одновременно пытаются схватить кусок, то он достаётся сильному крокодилу;
- если кусок мяса не попытался схватить ни один крокодил, то он сразу тонет и не достаётся никому;
- если крокодил заканчивает поедание ранее схваченного куска одновременно с бросанием нового, он будет пытаться его схватить.

Помогите Васе найти такие целые положительные A и B , чтобы модуль разности количества съеденного крокодилами мяса был минимальным. Учтите, что скорость поедания мяса каждым крокодилом составляет один килограмм в секунду.

Формат входных данных

В единственной строке задаётся количество килограммов мяса N ($2 \leq N \leq 10^9$) и период бросания кусков K ($1 \leq K \leq 10^9$). Гарантируется, что N — чётное число.

Все числа во входных данных целые.

Формат выходных данных

Размеры кусков A и B . Если существует несколько решений, выведите любое.

Примеры

стандартный ввод	стандартный вывод
4 3	1 3
4 1	2 2

Замечание

В первом тесте при любых A и B сильный крокодил съедает всё мясо.

Во втором тесте сильный крокодил хватается первый кусок, а слабому удаётся схватить второй, так как сильный в момент броска занят поеданием первого. Модуль разности количества съеденного мяса равен 0.