

## Задача А. Красивый номер

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

На берегу местной речки сидели два рыбака. Если присмотреться, в них можно было узнать Перуна-Нечитайло и Епифана Дормидонтовича Редисочкина. После того, как в Нью-Незамерзаенск приехал сити-менеджер Митрофан Хрушевелев, Редисочкин стал чувствовать себя, как он говорил, «почти английской почти королевой». Вот и вспомнил мэр свою давнюю страсть — рыбалку...

Что до Перуна-Нечитайло, то у него тоже появилось время на рыбалку — после первой же публикации про недостатки городского хозяйства вдруг выяснилось, что в местном бюджете совершенно нет денег на содержание газеты. И действительно, как бы их хватило на газету, если нужно срочно строить резиденцию сити-менеджера и гараж для трёх новых автомобилей, потому что сити-менеджер должен быстро перемещаться по городу.

А ещё сити-менеджер Хрушевелев собирается приобрести красивые (с его точки зрения) номера для своих трёх новых автомобилей. Хрушевелеву очень нравятся цифры  $a$  и  $b$ , и он считает красивыми номера, которые содержат только эти цифры. Номера автомобилей состоят из трёх цифр, так что выбор у Митрофана относительно небольшой.

Однако есть обстоятельство, которое дополнительно ограничивает Хрушевелева в выборе номеров. Хрушевелев обязан своим назначением Одному Большому Начальнику, и потому Митрофан полагает, что номера его автомобилей, если рассматривать их как числа, должны быть строго больше, чем самый большой номер автомобиля Одного Большого Начальника. Вместе с тем, Митрофан полагает, что номер тем лучше, чем он меньше, если рассматривать его как число.

Ваша задача — вывести в порядке возрастания три самых лучших красивых номера, которые может приобрести Хрушевелев. Если он может приобрести меньше трёх номеров, вместо недостающих номеров выведите  $-1$ .

### Формат входных данных

В первой строке содержатся две цифры  $a$  и  $b$  ( $0 \leq a, b \leq 9$ ) через пробел — цифры, которые нравятся Хрушевелеву. Гарантируется, что  $a \neq b$ .

Во второй строке содержатся три записанные подряд цифры — номер автомобиля Одного Большого Начальника.

### Формат выходных данных

Выведите три строки. В каждой из строк напечатайте по одному номеру из тех, которые может приобрести Хрушевелев. Номера должны быть упорядочены по возрастанию.

Если возможно приобрести меньшее количество номеров, чем три, выведите вместо недостающих номеров  $-1$ .

### Примеры

input.txt	output.txt
4 7 555	744 747 774
7 4 774	777 -1 -1
4 7 777	-1 -1 -1
0 1 001	010 011 100

## Задача В. Три шанса

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Рыба никак не обнаруживала своего существования в реке. Впрочем, пасмурное тихое осеннее утро располагало к неспешной беседе, и ничего плохого в том, что не нужно отвлекаться на удочки, рыбаки не видели. Редисочкин рассказывал про письмо от Потапа, который уехал учиться в магистратуру одного европейского университета.

Жажда знаний у Потапа была столь велика, что он записался едва ли не на все курсы, читавшиеся в текущем семестре, и теперь был порядком озадачен, как успеть выполнить все задания.

Для всех заданий существует два срока сдачи. Если студент успевает выполнить задание до истечения первого срока (включительно), он может получить полный балл за задание. Если в первый срок студент не уложился, но успел выполнить задание до истечения второго срока (включительно) — он может получить половину полного балла. Наконец, если он выполнит задание после истечения второго срока — он просто считается выполнившим задание.

Выполнять задания нужно обязательно: студент, не сдавший хотя бы одно задание по курсу, не будет допущен к экзамену. Важно также, что если студент выполнил каждое из заданий курса не менее, чем на половину полного балла, он может не сдавать экзамен, а получить оценку по результатам выполнения заданий.

Потап получил  $n$  заданий и определил для себя, в каком порядке он будет их выполнять. Отметим, что Потап будет выполнять задания последовательно, начиная с первого. К очередному заданию он будет приступать только тогда, когда выполнит предыдущее.

Считайте, что Потап начал выполнять задания утром в день #1. Если Потап закончил выполнять некоторое задание в день  $d$ , он может сдать это задание в этот день (поздно вечером). Но следующее задание он начнёт выполнять только на следующий день.

Ваша задача — выяснить, сколько и каких экзаменов Потапу придётся сдавать обязательно.

### Формат входных данных

В первой строке содержится целое число  $n$  ( $1 \leq n \leq 10^5$ ) — общее количество заданий в списке Потапа.

В каждой из следующих  $n$  строк содержится описание задания # $j$  ( $j = 1, 2, \dots, n$ ), состоящее из четырёх целых чисел:  $c_j, d_j^1, d_j^2, z_j$  ( $1 \leq c_j \leq 10^5, 1 \leq d_j^1 < d_j^2 \leq 10^6, 1 \leq z_j \leq 10^4$ ) — номера курса, к которому относится задание, первого и второго сроков сдачи и количества дней, которое потребуется Потапу на выполнение задания.

Задания перечислены в том порядке, в котором их будет выполнять Потап.

### Формат выходных данных

В первой строке выведите единственное целое число  $m$  — количество экзаменов, которые придётся сдавать Потапу.

Во второй строке выведите через пробел  $m$  целых чисел — номера курсов, по которым Потапу придётся сдавать экзамен, в порядке возрастания.

Если Потапу не придётся сдавать экзаменов, выведите как в первой, так и во второй строке 0.

## Примеры

input.txt	output.txt
10 22 5 12 4 17 8 11 4 2453 4 25 10 22 24 28 5 2453 20 42 8 17 20 25 1 2453 45 57 11 22 48 52 3 17 30 38 2 17 35 44 3	1 17
6 2241 3 8 2 5473 8 12 5 5473 14 20 3 2241 12 15 1 2241 14 18 1 2241 20 24 2	0 0
10 17342 12 28 12 15234 10 20 10 18532 11 25 10 15234 18 28 14 17342 20 32 12 18532 18 30 11 17342 22 35 8 15234 20 38 13 17342 32 38 11 18532 25 35 14	3 15234 17342 18532

## Замечание

Пояснение к первому примеру.

- В день #1 Потап приступит к выполнению задания по курсу #22 и завершит его в день #4. Поскольку первый срок сдачи задания — день #5, Потап получит за него полный балл.
- В день #5 Потап приступит к выполнению задания по курсу #17 и завершит его в день #8. Первый срок сдачи задания — день #8, за него Потап тоже получит полный балл.
- В день #9 Потап приступит к выполнению задания по курсу #2453 и завершит его в день #18. Первый срок сдачи задания — день #4 — Потап пропустит, но успеет сдать до второго срока — дня #25, и получит за это задание половину балла.
- В день #19 Потап приступит к выполнению задания по курсу #22 и завершит его в день #23. Поскольку первый срок сдачи задания — день #24, Потап получит за него полный балл.
- В день #24 Потап приступит к выполнению задания по курсу #2453 и завершит его в день #31. Первый срок сдачи задания — день #20 — Потап пропустит, но успеет сдать до второго срока — дня #42, и получит за это задание половину балла.
- В день #32 Потап приступит к выполнению задания по курсу #17 и завершит его в этот же день. Но и первый срок сдачи задания — день #20, и второй срок сдачи — день #25 — он уже пропустил, поэтому баллов за это задание он не получит. В этот момент становится понятно, что по курсу #17 Потапу уже точно придётся сдавать экзамен.

- В день #33 Потап приступит к выполнению задания по курсу #2453 и завершит его в день #43. Первый срок сдачи задания — день #45, так что Потап получит за это задание полный балл.
- В день #44 Потап приступит к выполнению задания по курсу #22 и завершит его в день #46. Это произойдёт до первого срока сдачи задания — дня #48, так что Потап получит за него полный балл.
- В день #47 Потап приступит к выполнению задания по курсу #17 и завершит его в день #48. Но и первый срок сдачи задания — день #30, и второй срок сдачи — день #38 уже пропущены, баллов Потап не получит.
- В день #49 Потап приступит к выполнению задания по курсу #17 и завершит его в день #51. Как и в предыдущем случае, Потап пропустил оба срока сдачи задания — дни #35 и #44, так что он вновь не получает баллов.

Итак, можно видеть, что Потап сдал все задания по курсу #22 на полный балл, а по курсу #2453 он получил за два задания половину балла, а за одно — полный. Что же касается курса #17, то три из четырёх заданий по нему Потап сдал после второго срока, поэтому ему придётся сдавать экзамен по этому курсу.

## Задача С. Короче...

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Перун-Нечитайло теперь зарабатывает написанием рекламных текстов. Бывают люди, которые сначала пишут короткие тексты, а потом уже добавляют в них подробности. Бывают люди, которые сначала пишут пространные тексты, а потом их сокращают. Перун-Нечитайло относится ко второму типу, и иной раз ему приходится трудно, когда текст должен иметь длину не более заданного количества знаков.

Текст, который написал Галактион, содержит  $m$  знаков, и теперь журналисту требуется сократить текст так, чтобы он содержал не более  $n$  знаков. Перун-Нечитайло хотел бы удалить из текста как можно меньшее количество слов. Но есть одна тонкость: некоторые слова в тексте являются ключевыми, и эти слова удалять нельзя.

Ради простоты будем считать, что текст содержит только слова, состоящие из строчных латинских букв, и слова разделяются ровно одним пробелом. Пробелы учитываются при подсчёте символов. В полученном тексте слова также должны разделяться ровно одним пробелом. И исходный, и полученный текст не начинаются и не заканчиваются пробелом. Длина каждого слова не превосходит 20 символов.

Ваша задача — определить, какое минимальное количество слов придётся удалить Галактиону.

### Формат входных данных

В первой строке содержатся целые числа  $m$ ,  $n$  и  $k$  ( $1 \leq n < m \leq 10^4$ ,  $1 \leq k \leq 10^2$ ) — количество символов в тексте Перуна-Нечитайло, требуемое количество символов в тексте и количество ключевых слов.

Далее следуют  $k$  строк, в каждой из которых записано по одному ключевому слову. Гарантируется, что все слова различны.

Далее следует текст Перуна-Нечитайло, начинающийся словом и заканчивающийся словом; все слова разделены не более чем одним пробелом.

Гарантируется, что длина любого из слов не превосходит 20 символов.

### Формат выходных данных

В первой строке выведите единственное целое число — минимальное количество слов, которое придется удалить Перуну-Нечитайло.

Если текст сократить нельзя, выведите в качестве ответа  $-1$ .

## Примеры

input.txt	output.txt
37 28 5 zadacha e ne takaya prostaya to be or not to be this is a question	1
37 27 5 ochevidnaya ideya ne daet ballov to be or not to be this is a question	2
37 28 4 spoillery ne horosho question to be or not to be this is a question	2
37 36 8 to be or not this is a question to be or not to be this is a question	-1
5 1 1 hello world	1

## Задача D. Игра

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В освободившихся помещениях газеты теперь открылся «Центр Развлечений ННнска». Поговаривали, что его владелец — то ли брат Хрушевелеву, то ли ещё какой родственник. Тех, кто заглядывал в центр развлечений, там всегда встречал немногословный молодой человек, представлявшийся Харитоном. Кто он в этом заведении — можно было только догадываться; впрочем, горожане предполагали, что Харитон как раз и является хозяином центра.

Особо любопытные посетители порой пытались расспросить Харитона — кто он, откуда приехал, и правда ли то, что про него говорят в городе. Харитон в таких случаях говорил, что готов ответить на любые вопросы, но только после того, как услышит правильные ответы на свои вопросы. А его вопросы были немного необычными.

Харитон предлагал посетителям сыграть с ним в следующую игру. У него есть  $n$  стаканов разного размера, при этом стакан с меньшим номером всегда помещается в стакан с большим номером.

Изначально все стаканы перевернуты вверх дном и стоят рядом. Харитон может взять какой-либо стакан и накрыть им любой другой стакан подходящего размера. Он проделывает это много раз, периодически спрашивая любопытного посетителя, под каким стаканом находится в настоящий момент стакан с некоторым номером  $\#j$ . Посетитель должен указать самый верхний из накрывающих стакан  $\#j$  стаканов (или собственно стакан  $\#j$ , если он не накрыт никаким другим).

Ваша задача — по заданной последовательности действий Харитона получить ответы на его вопросы.

### Формат входных данных

В первой строке содержатся целые числа  $n$  и  $q$  ( $1 \leq n \leq 2 \cdot 10^5$ ,  $1 \leq q \leq 4 \cdot 10^5$ ) — количество стаканов и общее количество действий: перекладываний стаканов и вопросов Харитона.

В каждой из следующих  $q$  строк описывается либо корректное перекладывание стакана, либо вопрос Харитона.

Описание перекладывания стакана начинается с числа 1, после которого через пробел следует сначала номер большего стакана, а затем — номер меньшего стакана, который будет накрыт большим. Гарантируется, что больший и меньший стаканы до перекладывания не накрыты никакими другими.

Описание вопроса начинается с числа 2, после которого через пробел следует номер стакана, местоположение которого интересует Харитона.

### Формат выходных данных

Выведите ответы на вопросы Харитона — по одному в каждой строке.

### Примеры

input.txt	output.txt
5 7	3
1 2 1	5
1 3 2	4
2 1	
1 5 4	
2 5	
1 5 3	
2 4	

### Замечание

Поясним приведённый пример.

- Первое действие — перекладывание стакана  $\#2$  на стакан  $\#1$ .

- Второе действие — перекладывание стакана #2 на стакан #1.
- Третье действие — вопрос, где находится стакан #1; правильный ответ — под стаканом #3.
- Четвертое действие — перекладывание стакана #5 на стакан #4.
- Пятое действие — вопрос, где находится стакан #5; правильный ответ — стакан #5 (он не накрыт никаким другим).
- Шестое действие — перекладывание стакана #5 на стакан #3.
- Седьмое действие — вопрос, где находится стакан #4; правильный ответ — стакан #4 (он не накрыт никаким другим).

## Задача Е. Утомлённые не солнцем

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Утренний туман над речкой понемногу рассеивался, кромки облаков подсвечивались солнцем, и ничто не мешало разговорам Редисочкина и Перуна-Нечитайло о том, как хорошо было бы, если бы Хрушевелев как-нибудь вскорости ушёл бы на повышение, а на его место назначили бы Потапа, когда он вернётся после учёбы...

За этими разговорами мы их и оставим, поскольку близится время разбора задач.

В состоявшемся соревновании было  $n$  задач, каждая из которых имела (по мнению жюри) сложность от 1 до 5. Теперь жюри предстоит провести разбор задач.

Конечно, жюри знает, что разбор утомляет участников (хотя и меньше, чем соревнование). Поэтому жюри хочет минимизировать утомлённость, которая накопится у участников после разбора. Утомлённость, накапливающаяся у участников после разбора, равна сумме утомлённостей от разбора каждой задачи.

Опытным путём жюри установило, что утомлённость от разбора некоторой задачи равна округлённой вниз полусумме сложности этой задачи и утомлённости от разбора предыдущей задачи.

Ваша задача — определить порядок, в котором жюри следует разбирать задачи. Считайте, что утомлённость участников перед началом разбора равна нулю.

### Формат входных данных

В первой строке содержится целое число  $n$  ( $1 \leq n \leq 100$ ) — количество задач в соревновании.

Во второй строке содержится  $n$  целых чисел  $c_j$  ( $1 \leq c_j \leq 5, j = 1, 2, \dots, n$ ) через пробел — сложности задач.

### Формат выходных данных

В первой строке выведите единственное целое число — значение утомлённости участников после разбора.

Во второй строке выведите  $n$  целых чисел через пробел — сложности задач в том порядке, в котором их следует разбирать.

### Примеры

input.txt	output.txt
5 1 2 4 1 5	6 1 1 2 4 5
7 4 2 2 4 2 4 1	11 1 2 2 2 4 4 4

### Замечание

Поясним первый пример.

Исходно утомлённость участников составляла 0.

- Разобрали задачу #1 сложности 1, утомлённость от этой задачи будет равна 0:  $\lfloor (0+1)/2 \rfloor = 0$ .
- Разобрали задачу #4 сложности 1, утомлённость от этой задачи также окажется равной 0:  $\lfloor (0+1)/2 \rfloor = 0$ .
- Разобрали задачу #2 сложности 2, утомлённость от этой задачи будет равной 1:  $\lfloor (0+2)/2 \rfloor = 1$ .
- Разобрали задачу #3 сложности 4, утомлённость от этой задачи примет значение 2:  $\lfloor (1+4)/2 \rfloor = 2$ .
- Разобрали задачу #5 сложности 5, утомлённость от этой задачи обратится в 3:  $\lfloor (2+5)/2 \rfloor = 3$ .

Суммарно получаем:  $0 + 0 + 1 + 2 + 3 = 6$ .