

## Задача А. Пик Балмера

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

*"Так называемый пик Балмера был открыт компанией Microsoft в конце 80-х. Неизвестно почему, но при концентрациях алкоголя в крови между 0,129 и 0,138% человек обретает сверхчеловеческие программистские способности. Однако, это довольно тонкий эффект, требующий тщательной калибровки. Вы не можете просто обеспечить кодеров годовым запасом виски и сказать им: "Программируйте!"*

([xkcd.ru/323/](http://xkcd.ru/323/))

Некоторая фруктовая компания для проведения испытаний и калибровки нового рецепта закупила несколько емкостей каждого из трех компонентов коктейля. Согласно рецепту, смешивать их нужно строго в определенной пропорции. Какое количество коктейля можно приготовить из имеющихся в наличии напитков?

### Формат входного файла

На первой строке даны три целых числа -  $a, b, c$  ( $1 \leq a, b, c \leq 500$ ) - количество каждого из трех компонентов коктейля соответственно.

На второй строке даны три целых числа -  $i, j, k$  ( $1 \leq i, j, k \leq 100$ ) - пропорции каждого из трех компонентов коктейля по рецепту (то есть, следует взять  $i$  частей первого напитка,  $j$  частей второго и  $k$  частей третьего).

### Формат выходного файла

Единственное число - максимальное количество коктейля, который можно приготовить из имеющихся в наличии напитков. Ответ будет засчитан как верный, если абсолютная или относительная погрешность не превосходит  $10^{-4}$ .

### Примеры

stdin	stdout
10 10 10 1 1 1	30.0
10 10 10 1 2 3	20.0

## Задача В. Упражнение

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Учительница задала своему классу на лето большое задание на сложение. Каждому ученику даны два числа -  $n$  и  $m$ . Надо выписать первые  $n$  натуральных чисел в ряд, а затем  $m$  раз проделать следующую операцию: сначала удалить каждое  $m + 1$ -е число (стоящие на позиции  $m + 1, 2(m + 1), \dots$ ), а затем заменить оставшиеся префиксными суммами (например, вместо "1 2 3 4" получится "1 3 6 10"), затем удалить каждое  $m$ -е число, и так далее. Учительница рассчитывает надолго занять ребят, но кому хочется летом решать большие задачи по арифметике? Помогите ребятам!

### Формат входного файла

В единственной строке входного файла даны два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 10^6$ ).

### Формат выходного файла

Выведите получившиеся числа по одному в каждой строке по модулю 1000000007.

### Примеры

stdin	stdout
4 4	1
4 1	1 4

## Задача С. Радиостанция

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Радиостанция ежедневно проигрывает большой плейлист, который состоит из избранных экспертами треков. Длина этого плейлиста -  $n$ , количество различных избранных треков -  $m \leq n$ . Хорошим считается плейлист, в котором каждые  $m$  подряд идущих треков различаются. Т.е. когда бы ни подключился слушатель - он всегда услышит сначала все отобранные треки без повторов.

Необходимо по заданному плейлисту определить, является ли он хорошим.

### Формат входного файла

В первой строке даны через пробел числа  $n, m$  ( $1 \leq n \leq 10^5, 1 \leq m \leq n$ ) - длина плейлиста и количество различных отобранных треков соответственно. В следующей строке заданы  $n$  номеров треков через пробел ( $1 \leq k_i \leq m$ ).

### Формат выходного файла

Если данный плейлист хороший - выведите в единственной строке "Yes" иначе выведите "No". Выводить следует без кавычек.

### Примеры

stdin	stdout
4 2 1 2 1 2	Yes
4 3 1 2 3 2	No

### Note

Во втором тесте пользователь подключившийся во время второго трека услышит их в порядке "2 3 2" значит данный плейлист плохой.

## Задача D. Дороги

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 4 секунды  
Ограничение по памяти: 64 мегабайта

В государстве есть набор из  $n$  пунктов остановки, связанных  $m$  дорогами определенной длины. Дороги двунаправленные и также возможно существование нескольких дорог из одного пункта в другой. Из  $n$  пунктов остановки только  $k \leq n$  являются городами. Путешественники хотят быстро преодолевать расстояние между городами в государстве, а чтобы сделать пути еще короче - постоянно строятся новые дороги между городами.

Вам задана начальная карта дорог и набор пунктов остановки, являющихся городами. Необходимо отвечать на запросы двух типов. Первый тип запроса это запрос расстояния между двумя городами, второй - добавление новой дороги между какими-либо двумя городами. Обратите внимание, что все запросы выполняются только для пунктов остановки, являющихся городами.

### Формат входного файла

В первой строке заданы числа  $n$  и  $m$  ( $2 \leq n \leq 1000, 0 \leq m \leq 10^5$ ) через пробел - количество пунктов остановки и количество дорог в изначальной конфигурации. В последующих  $m$  строках описаны дороги. Описание дорог состоит из трех чисел  $a, b$  и  $len$  ( $1 \leq a, b \leq n, a \neq b, 1 \leq len \leq 10^6$ ), разделенных пробелом - номера соединяемых дорогой пунктов остановки и длина дороги.

В следующей строке задано единственное число  $k$  ( $2 \leq k \leq 100$ ) - число пунктов остановки, являющихся городами. Далее в строке заданы через пробел номера этих пунктов  $num_i$  ( $1 \leq num_i \leq n$ ) без повторов.

В следующей строке задано единственное число  $q$  ( $1 \leq q \leq 10^4$ ) - количество запросов. В последующих  $q$  строках заданы описания запросов. Запросы бывают двух типов: "1  $a$   $b$ " и "2  $a$   $b$   $len$ " ( $1 \leq a, b \leq n, a \neq b, 1 \leq len \leq 10^6$ ) - первое число означает тип запроса, далее идут номера городов и при втором типе длина добавляемой дороги.

Все данные длины дорог целочисленные.

### Формат выходного файла

Для каждого запроса первого типа выведите в отдельную строку длину пути между заданными в запросе городами. Если пути не существует, выведите -1.

### Примеры

stdin	stdout
2 1 1 2 10 2 1 2 1 1 1 2	10
2 0 2 1 2 5 1 1 2 2 1 2 2 1 2 1 2 2 1 1 1 1 2	-1 2 1

## Задача Е. Замечательное решение

Имя входного файла:	<code>stdin</code>
Имя выходного файла:	<code>stdout</code>
Ограничение по времени:	15 секунд
Ограничение по памяти:	128 мегабайт

Маюши опять попала в беду! Чтобы спасти ее, Ринтаро придется вмешаться в ход истории. Однако менять прошлое надо крайне осторожно и чтобы не случилось непоправимого, вам надо написать программу, которая будет следить за действиями Ринтаро и контролировать его.

Математически состояние вселенной описывается в виде квадратной Матрицы Мира со стороной  $n$  ( $1 \leq n \leq 500$ ). Устройство, доступное Ринтаро, позволяет изменить за один раз от одного до трех элементов матрицы. Иногда Ринтаро хочет убедиться, что сделал все правильно и проверяет значения некоторых фрагментов Матрицы. Когда он понимает, что совершил слишком много ошибок, он может откатить Матрицу к виду, который она имело до одной из предыдущих операций изменения элемента.

Помогите спасти Маюши!

### Формат входного файла

В первой строке дано единственное целое число  $n$  ( $1 \leq n \leq 500$ ).

В каждой из следующих  $n$  строк дано по  $n$  целых положительных чисел, значение каждого из которых не превышает  $10^9$ .

На следующей строке дано число  $m$  ( $1 \leq m \leq 100000$ ) - количество действий Ринтаро.

Каждая из следующих  $m$  строк содержит описание его действий.

Если строка начинается с символа "i" то Ринтаро планирует внести прямые изменения в Матрицу Мира. В таком случае после символа и пробела в строке дается целое число  $a$  ( $1 \leq a \leq 3$ ) - количество изменяемых элементов, за которым следуют описания этих элементов в группах по три целых числа  $i, j, k$  ( $1 \leq i, j \leq N, 1 \leq k \leq 10^9$ ) - строка и столбец матрицы и новое значение элемента соответственно.

Если строка начинается с символа "f" то Ринтаро решил восстановить старую Матрицу Мира. После символа и пробела в строке следует единственное целое число  $i$  ( $1 \leq i \leq m$ ), которое показывает номер первой операции изменения (операцией изменения считается запрос типа "i" или "f"), после которой, по мнению Ринтаро, возникла ошибка. В таком случае Матрица должна быть возвращена к виду, который она имела до  $i$ -ой модификации элементов. Гарантируется, что на момент получения этого запроса произошло не меньше  $i$  операций изменения.

Если строка начинается с символа "с" то Ринтаро решил проверить, что находится в матрице. После символа и пробела в строке указаны четыре целых числа  $x_1, y_1, x_2, y_2$  ( $1 \leq x_1 \leq x_2 \leq n, 1 \leq y_1 \leq y_2 \leq n$ ) - координаты противоположных элементов проверяемой подматрицы. Ринтаро хочет проверить контрольное значение в этом прямоугольнике. Контрольное значение вычисляется как побитовое исключающее ИЛИ (XOR) элементов данной подматрицы.

### Формат выходного файла

Каждому запросу типа "с" должна соответствовать строка, содержащая единственное целое число - контрольное значение подматрицы, которую проверяет Ринтаро.

## Примеры

stdin	stdout
2	2
1 2	30
4 8	16
12	8
c 1 2 1 2	2
i 1 1 1 16	1
c 1 1 2 2	30
c 1 1 1 1	15
i 2 1 1 8 2 2 1	
c 1 1 1 1	
c 1 2 1 2	
c 2 2 2 2	
f 2	
c 1 1 2 2	
f 3	
c 1 1 2 2	

## Note

В примере: в результате первого запроса выводится значение элемента с координатами (1, 2). После первого изменения матрица принимает вид [16 2];[4 8]; после второго изменения матрица принимает вид [8 2];[4 1]; после третьего (откат к состоянию перед вторым изменением) [16 2];[4 8]; после четвертого (откат к состоянию перед третьим изменением) [8 2];[4 1];.

## Задача F. Склад

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Склад в порту загроможден контейнерами. Перед проверкой необходимо срочно навести порядок, причем как можно быстрее.

Контейнеры стоят строго по направляющим линиям, которые представляют собой квадратную сетку размером  $n$  на  $n$ . Изначально контейнеры могут стоять друг на друге. После уборки каждый контейнер должен стоять на полу, причем все контейнеры должны находиться в прямоугольной области и в каждой ячейке сетки должен стоять ровно один контейнер (то есть, они должны быть плотно упакованы в ровный прямоугольник).

Любой контейнер можно переместить из любого положения в любое свободное место на полу за один час. Определите минимальное время, которое займет перестановка, если в каждый момент времени перемещать можно только один контейнер.

### Формат входного файла

В первой строке даны два целых числа  $n$  и  $m$  ( $1 \leq n \leq 100, 1 \leq m \leq n^2$ ) - размер сетки и количество контейнеров на складе.

В каждой из следующих  $m$  строк даны два целых числа  $x_i$  и  $y_i$  ( $1 \leq x_i, y_i \leq n$ ) - координаты каждого из контейнеров.

### Формат выходного файла

Выведите единственное число - минимальное количество перемещений. Гарантируется, что решение всегда существует.

### Примеры

stdin	stdout
3 2 1 1 1 1	1
4 3 2 2 1 1 4 4	2

## Задача G. Двадцать одно

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Широко известна игра под названием "Двадцать одно" это локализованная версия американского "BlackJack" который является самой популярной игрой в казино по всему миру. Также широко известна история про студентов, которые в течении длительного времени постоянно выигрывали крупные суммы с помощью метода подсчета карт. Вам задана упрощенная версия правил и фактически требуется повторить достижение этих известных студентов. Одним из главных отличий является то, что игроки знают карты в колоде и карты соперника, что позволяет выстроить абсолютно оптимальную стратегию.

Во входном файле заданы числа  $a$  и  $b$  - количество уже набранных очков первым и вторым игроками соответственно. Также дана колода из  $n$  карт, каждая из которых имеет стоимость от 1 до 11 очков. Игроки по очереди достают карты из колоды, при этом стоимость полученных карт суммируется с уже набранными очками. Можно считать что карты из колоды достаются независимо и равновероятно. Если в какой-то момент один из игроков набирает больше чем 21 очко - он проигрывает, а его противник соответственно выигрывает. Также игрок может на своем ходу отказаться брать карту, если считает что ему хватит, но после этого он уже не сможет взять ни одной карты и будет просто пропускать ход. Если оба игрока прекратили брать новые карты выигрывает тот, кто набрал большее количество очков. Поскольку второй игрок является дилером - он решил увеличить себе шансы правилами и при равенстве очков присуждать победу себе. Также гарантируется что игра не может закончиться тем, что доступные карты в колоде закончатся раньше, чем кто либо переберет больше 21 очка.

Оба игрока очень умны и обладают всей доступной информацией, поэтому играют оптимально. Под оптимальностью подразумевается то, что в каждой ситуации игрок выполняет действие, приносящее ему максимальную вероятность победы при также оптимальной игре соперника. Вы играете на месте первого игрока, ваша задача определить вероятность вашей победы при заданных условиях.

### Формат входного файла

В первой строке через пробел даны числа  $a, b, n$  ( $0 \leq a, b \leq 21, 1 \leq n \leq 13$ ). Во второй строке через пробел заданы  $n$  целых чисел  $val_i$  ( $1 \leq val_i \leq 11$ ) - стоимости карт, лежащих в колоде.

### Формат выходного файла

Необходимо вывести вероятность победы первого игрока. Ответ будет считаться верным, если абсолютная или относительная погрешность не превосходит  $10^{-6}$ .

### Примеры

stdin	stdout
20 20 3 1 1 2	0.3333333333333333
21 19 3 1 2 3	0.6666666666666666

### Note

В первом примере равное количество очков, поэтому если первый игрок не будет брать карт - второй тоже не будет, и первый точно проиграет. Поэтому придется взять карту, если попадется карта весом 2 - сразу проигрываем. В 2 случаях из трех попадется 1, и мы наберем максимальный балл - 21. После этого второй игрок попытается победить, и для этого возьмет еще одну карту, с вероятностью  $1/2$  он также получит 21 очко и мы проиграем. Итоговая вероятность победы равна  $2/3 * 1/2 = 1/3$ .