

## Задача А. Разноцветная фотография

Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

В ряд стоят  $n$  ЛКШат. Каждый стоит в ЛКШатской футболке какого-то цвета, соответствующего году проведения ЛКШ. Культорги хотят выбрать какой-то подотрезок из ЛКШат, чтобы сфотографировать их.

*Разноцветностью* фотографии культорги называют количество различных цветов футболок на фотографии.

Так как культорги ещё не определились, какой подотрезок ЛКШат выбрать для фотографии, они просят вас посчитать сумму разноцветностей фотографий по всем возможным подотрезкам ЛКШат. Помогите им с этим!

### Формат входных данных

В первой строке записано целое число  $n$  — количество ЛКШат в ряду ( $1 \leq n \leq 100\,000$ ).

Во второй строке записаны  $n$  целых чисел  $a_i$  — цвета футболок ЛКШат в порядке их следования в ряду ( $1 \leq a_i \leq 100\,000$ ).

### Формат выходных данных

В единственной строке выведите ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
5 4 2 3 4 4	29
1 1	1

## Задача В. А ты растаможь слона!

Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Как известно, работники таможи пропускают через границу груз, только если он соответствует определенным условиям. В этот раз эстонские таможенники прознали о проведении Летней Компьютерной Школы неподалеку от Нарвы.

Специально для ЛКШат они придумали следующее условие пропуска груза: в одном автобусе можно провезти только коробки, веса которых не делятся друг на друга. Оргкомитет ЛКШ обнаружил  $2n$  коробок с блокнотами, ручками, футболками и спортивным инвентарем на своей конспиративной квартире в Санкт-Петербурге. Веса всех коробок различны и равны от 1 до  $2n$ , соответственно. Они хотят положить в один автобус как можно больше коробок.

Помогите им узнать, какое максимальное количество коробок можно положить в один автобус, чтобы он беспрепятственно прошел эстонский таможенный контроль.

### Формат входных данных

В первой строке входных данных записано единственное число  $n$ , равное половине числа обнаруженных коробок ( $1 \leq n \leq 100$ ).

### Формат выходных данных

В единственной строке выведите единственное число, равное максимальному количеству коробок, которое можно загрузить в один автобус так, чтобы он беспрепятственно пересёк границу.

### Примеры

стандартный ввод	стандартный вывод
3	3

### Замечание

Для примера одним из возможных ответов являются веса: 2, 3, 5.

## Задача С. «Дрыц-тыц телепатик»

Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Компания связи «Дрыц-тыц телепатик» планирует провести сеть телепатической связи в ЛКШляндии. Для этого компании нужна система номеров телепатических телефонов.

Ваша задача — по данному населению городов ЛКШляндии придумать уникальный код для каждого города так, чтобы всевозможных номеров хватило на всех жителей. Номер вместе с кодом состоит из 10 цифр, причём может начинаться с нуля. Номер начинается с непустого кода города и имеет хотя бы одну цифру в части после кода. Например, если в городе Культоргов используют код 045, то телефон Арины может быть таким: (045) 192-01-51.

Современные стандарты безопасности телепатической связи также требуют, чтобы коды городов обладали свойством *беспрефиксности*: никакой код города не может являться префиксом (началом) кода другого города.

### Формат входных данных

В первой строке написано одно целое число  $n$  — количество городов в ЛКШляндии ( $1 \leq n \leq 100\,000$ ). В следующей строке записано  $n$  целых числа  $a_i$  — населения городов ( $1 \leq a_i \leq 10^9$ ).

### Формат выходных данных

Выведите  $n$  строк: для каждого города в ЛКШляндии выведите код для него на отдельной строке. Коды городов необходимо выводить в том же порядке, в котором были даны города во входном файле. Если возможных наборов кодов несколько, выведите любой из них.

Если коды, удовлетворяющие условию, выбрать невозможно, выведите «Impossible» (без кавычек).

### Примеры

стандартный ввод	стандартный вывод
3 8372000 165 123456789	44 1234567 0
11 166 1000000000 1000000000 1000000000 1000000000 1000000000 1000000000 1000000000 1000000000 1000000000 1000000000	Impossible

## Задача D. Кружок рисования

Ограничение по времени: 1 секунда (python3: 3 секунды)

Ограничение по памяти: 256 мегабайт

В один из дней на клубе рисования в ЛКШ изучалось современное искусство. Картины в направлении «кукарекизма» представляют собой длинную полоску бумаги, разбитую на ячейки, которые покрашены разными цветами. Будем считать, что каждый цвет можно представить как целое неотрицательное число.

Картины в этом жанре принято рисовать следующим образом: выбирается цвет краски и закрашиваются все ячейки от начала полоски до выбранной ячейки с номером  $d$ .

Длина полоски  $10^9$  ячеек, поэтому ходить вдоль картины всем лень. Напишите программу, которая подскажет ещё в процессе рисования, в какой цвет покрашена ячейка.

Изначально все ячейки покрашены в цвет 0.

### Формат входных данных

Первая строка входных данных содержит одно число  $q$  ( $1 \leq q \leq 100\,000$ ) — количество запросов.

Следующие  $q$  строк описывают действия. Каждая строка начинается с числа  $type$ . Если  $type = 1$ , то дальше идут ещё 2 числа  $r$  и  $c$  ( $1 \leq r, c \leq 10^9$ ), это обозначает, что красятся первые  $r$  ячеек в цвет  $c$ . Если же  $type = 2$ , то дальше идёт число  $x$ , это обозначает, что спрашивается, в какой цвет окрашена ячейка с номером  $x$ .

### Формат выходных данных

Выведите в отдельных строках ответы на запросы, в какой цвет покрашена очередная ячейка.

### Примеры

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

## Задача E. В поисках мусорки

Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

### Это интерактивная задача.

Робот R8D16 накопил за день довольно много мусора и теперь хочет его выкинуть. К сожалению, он совсем забыл, где находится мусорка, и теперь хочет это узнать.

Будем считать, что робот находится на плоскости с введенной на ней прямоугольной системой координат. Мусорка расположена в точке с целыми координатами  $(x_0, y_0)$ . R8D16 точно знает, что корзина расположена не слишком далеко от центра координат, а именно, выполняется неравенство  $-10^9 \leq x_0, y_0 \leq 10^9$ .

Робот может посылать запросы на специальный сервер, отслеживающий положение всех объектов на плоскости. В качестве запроса R8D16 может послать на сервер целочисленные координаты точки на плоскости, а сервер в ответ отправит роботу *манхэттенское* расстояние от этой точки до точки, в которой расположена мусорка. Манхэттенским расстоянием между двумя точками на плоскости называется сумма абсолютных значений разностей координат этих точек по осям  $x$  и  $y$ . Иными словами, если R8D16 передает на сервер координаты точки  $(x, y)$ , в ответ он получит значение величины  $|x - x_0| + |y - y_0|$ . В силу особенностей работы сервера, для всех запросов должно выполняться неравенство  $-10^9 \leq x, y \leq 10^9$ .

R8D16 хочет как можно скорее узнать, где же располагается мусорка. Помогите ему сделать это, потратив не более трёх запросов!

### Протокол взаимодействия

После начала работы программа должна вывести несколько запросов к серверу. Чтобы послать на сервер запрос, содержащий точку  $(x, y)$ , программа должна вывести строку «? x y». После этого ваша программа должна считать ответ на запрос — манхэттенское расстояние до мусорки от точки, указанной в запросе.

Когда ваша программа определит положение мусорки, она должна вывести строку «! x<sub>0</sub> y<sub>0</sub>» — координаты точки, в которой располагается корзина. После этого ваша программа должна немедленно завершиться.

### Примеры

стандартный ввод	стандартный вывод
5	? 0 0
	? 1 1
3	? 3 2
2	! 2 3

### Замечание

Вывод **каждой строки** должен завершаться переводом строки и сбросом буфера потока вывода. Для этого используйте `flush(output)` на языке Pascal, `fflush(stdout)` в C/C++ или `cout.flush()` в C++, `sys.stdout.flush()` на языке Python, `System.out.flush()` на языке Java.

## Задача F. Ночь с перфоратором

Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Приехав на новую базу, админ Паша обнаружил, что в комповнике отсутствует сеть.

Он планирует оборудовать в комповнике  $n$  розеток, соединенных проводами. Для каждого провода известно, какие розетки он соединяет, а также его пропускная способность. Сеть является связной, а также не имеет в себе циклов, так как при их наличии возникнет замыкание.

Пропускная способность пути из нескольких проводов представляет собой побитовое «И» их пропускных способностей, так как каждый из проводов представляет собой 30 бит, некоторые из которых не перебиты.

Паша хочет найти путь с максимальной пропускной способностью в сети. Мы не просим вас находить этот путь, но просим вас найти величину его пропускной способности.

### Формат входных данных

В первой строке входных данных задано число  $n$  — количество розеток в комповнике ( $2 \leq n \leq 100\,000$ ).

В последующих  $n - 1$  строках содержатся числа  $a_i, b_i, c_i$  — номера розеток, которые соединяет провод, и его пропускная способность. ( $1 \leq a_i, b_i \leq n, 1 \leq c_i \leq 10^9$ ).

### Формат выходных данных

В единственной строке выходных данных выведите число — ответ на поставленную задачу.

### Примеры

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

### Замечание

Побитовое «И» — это бинарная операция, действие которой эквивалентно применению логического «И» к каждой паре битов, которые стоят на одинаковых позициях в двоичных представлениях операндов (Материал из Википедии — свободной энциклопедии). В языках программирования обычно представляется оператором `&`.

## Задача G. Поезд вокруг Костромы

Ограничение по времени: 0.5 секунды (python3: 5 секунд)

Ограничение по памяти: 256 мегабайт

Аля Игорьева отправилась в ЛКШ под Кострому на новом фирменном поезде “Иван Сусанин”. К сожалению, поезд оказался зациклен. Чтобы скоротать время в пути, Аля решила посчитать среднюю температуру в поезде. Температуры в вагонах различны в силу отличающихся настроек систем кондиционирования в вагонах.

Аля идет по поезду по часовой стрелке, начиная с какого-то вагона. В каждом вагоне она останавливается, замеряет температуру и записывает среднюю температуру по всем уже пройденным вагонам на бумажку. Она знает длину поезда, поэтому проходит по каждому вагону ровно один раз.

В силу сложившихся обстоятельств, Аля забыла, с какого вагона она начала обходить поезд. Она знает лишь, что все, записанные ею на бумажку числа, положительные. Помогите ей узнать количество вагонов, с которых она могла начать свой обход.

### Формат входных данных

В первой строке вводится число  $n$  — длина поезда ( $1 \leq n \leq 500\,000$ ).

В следующей строке вводится  $n$  чисел  $a_i$  — температура в  $i$ -м вагоне, начиная с какого-то вагона по часовой стрелке ( $|a_i| \leq 10^9$ ).

### Формат выходных данных

Выведите одно число — количество вагонов, с которых могла начать Аля.

### Примеры

стандартный ввод	стандартный вывод
5 18 22 16 28 20	5
6 18 20 22 19 -57 -196	0

### Замечание

Во втором примере вагон номер 5 — вагон-рефрижератор, вагон номер 6 — цистерна с жидким азотом.

## Задача Н. Яблоки и бананы

Ограничение по времени: 0.3 секунды (python3: 3 секунды)

Ограничение по памяти: 256 мегабайт

Культурги для одного из мероприятий распечатали  $n$  волшебных букв, которые превращаются в то, что на них падает. Буквы разложили в ряд под крыльцом ГК так, чтобы они образовали последовательность  $s$ .

Преподаватель Андрей на полднике в ЛКШ набрал  $q$  фруктов в коробку. Оказалось, что там лежат только яблоки и бананы. Андрей начал нести эту коробку из ресторана на улицу, и тут у него развязались шнурки. Он поставил коробку на край крыльца, начал завязывать шнурки. Внезапно подул сильный ветер и яблоки с бананами полетели вниз.

К счастью, внизу лежала последовательность букв. И, когда яблоко падало вниз, то начиная с какой-то позиции  $pos$  пять букв превращались в слово «apple». А когда на буквы падал банан, то начиная с какой-то позиции  $pos$  шесть букв превращались в слово «banana». Если до конца последовательности оставалось меньше букв, чем в слове «apple» или «banana», они превращались в соответствующее начало этого слова.

Андрея впечатлили падающие яблоки и бананы, и он, вместо того, чтобы ловить их, следил за процессом и пытался подсчитать, сколько названий фруктов «apple» и «banana» можно прочитать в этой последовательности букв. Какие числа у него получились?

### Формат входных данных

В первой строке заданы два числа  $n$  и  $q$  — длина строки  $s$  и количество фруктов  $q$  ( $1 \leq n, q \leq 10^4$ ).

Во второй строке задана последовательность букв  $s$ . Все символы последовательности  $s$  являются строчными латинскими буквами.

В следующих  $q$  строках задаются два числа  $t_i$  и  $pos_i$  ( $1 \leq t_i \leq 2, 1 \leq pos_i \leq n$ ). Если  $t_i$  равно 1, то начиная с позиции  $pos$  падает «apple», если  $t_i$  равно 2, то начиная с позиции  $pos$  падает «banana».

### Формат выходных данных

Выведите  $q + 1$  строку. В первой строке выведите, сколько названий фруктов в последовательности  $s$  в самом начале. В  $(i + 1)$ -й строке ( $i \geq 1$ ) выведите, сколько названий фруктов после  $i$ -го падения фрукта.

### Примеры

стандартный ввод	стандартный вывод
10 5 bananapple	2
1 1	2
2 5	1
1 4	1
2 4	1
1 3	1
7 1 abacaba	0
1 4	0

### Замечание

В первом примере строка модифицируется следующим образом:

bananapple → appleapple → applbanana → appapplena → appbananaa → appplena.



## Задача I. Как склеить осу

Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Однажды на полднике ЛКШнок Игнат взял абрикос, положил его на землю и намазал клеем. В итоге на него слетелось много-много ос. Абрикос они, конечно, съели, но в ходе этого они случайно склеились в одну большую осу. Игната это позабавило, и он побежал рассказывать своим друзьям про необычную большую осу. Но он забыл, как именно она выглядела.

Игнат запомнил лишь то, что большая оса состояла из  $n$  ос,  $m$  пар из которых были соединены клеем. Причем Игнат запомнил, что шанс вырваться из этого ада был у  $k$  ос: они были соединены ровно с одной другой осой. Также он заметил, что большую осу нельзя было разделить на части, не разрывая клей.

Помогите Игнату вспомнить, как именно выглядела большая оса.

### Формат входных данных

В первой строке написаны три числа:  $n$  ( $1 \leq n \leq 100\,000$ ),  $m$  ( $0 \leq m \leq 100\,000$ ) и  $k$  ( $0 \leq k \leq 100\,000$ ).

### Формат выходных данных

Выведите в первой строке «Yes», если такая большая оса могла существовать, и «No», если Игнат что-то перепутал, и такой большой осы не существует.

Если большая оса существует, то необходимо вывести какую-нибудь конфигурацию большой осы. В следующих  $m$  строках после «Yes» выведите по паре разных чисел от 1 до  $n$  — номера ос, которые соединены клеем в этой конфигурации. Пары не могут повторяться.

Если конфигураций несколько, можно вывести любую из них.

### Примеры

стандартный ввод	стандартный вывод
6 7 2	Yes 1 3 2 6 3 4 4 5 5 6 3 5 3 6
5 4 1	No

### Замечание

В первом примере приводится конфигурация из шести ос, из них соединены клеем ровно семь пар. Шанс вырваться есть только у двух ос: под номером 1 (она соединена только с осой 3) и под номером 2 (она соединена только с осой 6).

Во втором примере такой конфигурации не существует.

## Задача J. Набор двумя пальцами

Ограничение по времени: 0.3 секунды (python3: 16 секунд)

Ограничение по памяти: 256 мегабайт

У ЛКШонка Васи есть клавиатура, состоящая из четырёх рядов клавиш длины  $s$  каждый. Каждая клавиша содержит некоторый символ, все символы различны.

Вася — ярый адепт секты «печатать двумя пальцами», он собирается набрать строку  $s$ . Чтобы перенести палец с позиции  $c_1$  ряда  $r_1$  в позицию  $c_2$  ряда  $r_2$ , Вася тратит  $|r_1 - r_2| + |c_1 - c_2|$  энергии. Изначально Васины два пальца находятся в противоположных углах первого ряда, помогите ему набрать сообщение, затратив минимум энергии.

### Формат входных данных

В первой строке входных данных записано единственное число  $c$  — длина клавиатуры ( $1 \leq c \leq 1000$ ).

Следующие четыре строки содержат описание рядов клавиатуры, по  $c$  целых чисел от 1 до  $4c$ . Гарантируется, что все  $4c$  чисел различны.

Следующая строка содержит  $n$ , длину строки  $s$  ( $1 \leq n \leq 1000$ ).

В последней строке  $n$  целых чисел от 1 до  $4c$  — строка  $s$ .

### Формат выходных данных

В первой строке выведите минимальный расход энергии на печать строки  $s$ .

Во второй строке выведите  $n$  чисел 1 и 2,  $i$ -е число обозначает, каким пальцем нажимать  $i$ -ю букву. Если оптимальных ответов несколько, выведите любой.

### Примеры

стандартный ввод	стандартный вывод
5	10
1 2 3 4 5	2 2 2 2 2 2 2 2 2 2 2
10 9 8 7 6	
15 14 13 12 11	
16 17 18 19 20	
12	
11 12 13 14 15 15 15 15 11 11 11 11	

### Замечание

Первым во вводе идёт именно первый ряд (ряд, на котором стоят Васины пальцы).

Палец номер 1 стоит в слева, палец номер 2 стоит справа. Длина рук, пальцев, и другие физические характеристики позволяют Васе без проблем дотянуться любым пальцем до любой части клавиатуры.

## Задача К. Камни на море

Ограничение по времени: 1 секунда (python3: 4 секунды)

Ограничение по памяти: 256 мегабайт

Ваня и Иоанна вышли на берег Балтийского моря. Когда солнце уже садилось, им в глаза бросилось странное явление. На песке была разложена бесконечная линия из камней, у которой есть начало, но нет конца. Сами камни были пронумерованы слева направо, начиная с единицы.

Иоанна испугалась такого перфекционизма, но Ваня хотел показать свой характер и решил посчитать НОД (наибольший общий делитель) и НОК (наименьшее общее кратное) всех чисел на любом отрезке, который назовет Иоанна. Чтобы Иоанна не испугалась второй раз, Ваня будет показывать только последнюю цифру каждого из полученных чисел.

### Формат входных данных

В первой строке дано количество отрезков  $q$  ( $1 \leq q \leq 50\,000$ ), про которые спрашивает Иоанна.

В  $i$ -й из следующих  $q$  строк написано по два числа  $left_i$  и  $right_i$  ( $1 \leq left_i \leq right_i \leq 10^9$ ), границы отрезка камней, про который спрашивает Иоанна.

### Формат выходных данных

Выведите  $q$  строк. В  $i$ -й строке выведите сначала последнюю цифру НОД всех номеров на отрезке камней  $[left_i, right_i]$ , а затем через пробел последнюю цифру НОК всех номеров на этом отрезке.

### Примеры

стандартный ввод	стандартный вывод
4	5 5
5 5	1 0
8 10	1 0
12 15	1 2
1 4	
3	1 0
4 5	1 0
10 20	1 6
11 13	