

## Задача А. Attention! This is Div C/D!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

Если вы видите эту задачу в качестве первой задачи тура, то вы в дивизионе C/D. В случае, если это не тот дивизион, в котором вы рассчитывали участвовать — немедленно перезайдите в нужный дивизион!

«DIVCD» — это не только обозначение дивизиона, но и строка из римских цифр длины 5. Самое длинное римское число в этой строке (DIV=504) имеет длину 3. Но участник Div C/D должен уметь переходить от частной задачи к общей.

Строка нам не дана. Задана только длина строки  $n$ , составленной из букв I, V, X, L, C, D и M. Найти среднюю длину самого длинного римского числа, которое является подстрокой этой строки.

Напоминаем современный способ записи римских чисел (таблица взята из Википедии):

Значение разряда	Тысячи	Сотни	Десятки	Единицы
1	M	C	X	I
2	MM	CC	XX	II
3	MMM	CCC	XXX	III
4		CD	XL	IV
5		D	L	V
6		DC	LX	VI
7		DCC	LXX	VII
8		DCCC	LXXX	VIII
9		CM	XC	IX

Заметим, что:

- Числа 4, 9, 40, 90, 400 и 900 записываются в реверсивной нотации, где первый символ вычитается из второго (например, для 40 (XL) ‘X’ (10) вычитается из ‘L’ (50)). Это **единственные** места, где реверсивная нотация используется.
- Число, содержащее несколько десятичных цифр, строится дописыванием римского эквивалента каждой цифры от старшего разряда к младшему.
- Если в десятичном разряде стоит 0, никаких цифр в этом разряде в римском представлении не пишется.
- Наибольшее число, которое может быть представлено в римской системе счисления — это число 3,999 (MMMCMXCIX).

### Формат входных данных

Входные данные содержат одно целое число  $n$  ( $1 \leq n \leq 1000$ ) — длину строки.

### Формат выходных данных

Можно показать, что ответ единственным образом представляется в виде рационального числа  $p/q$ , где  $p$  и  $q$  — взаимно простые целые положительные числа. Выведите число  $p \cdot q^{-1} \bmod 998\,244\,353$ , то есть остаток от деления на 998 244 353 произведения  $p$  на такое число  $q'$ , что  $q \cdot q' - 1$  делится на 998 244 353.

### Примеры

стандартный ввод	стандартный вывод
2	509308345
1	1

## Задача В. Буквы в стеках

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 1024 мегабайта

Добро пожаловать в дивизион АВ!

Двое игроков играют в игру. Изначально у них есть три непустых стека, в которых лежат буквы, причем буквы могут быть только «А» и «В». Известно также, что ни в каком стеке нет двух соседних одинаковых букв. Игроки делают ходы по очереди. В свой ход игрок должен взять верхнюю букву любого непустого стека и положить её на вершину другого стека такого, что до перемещения на его вершине лежала такая же буква. После этого две этих равных буквы удаляются с вершины стека, на который положили букву. Ваша задача – определить, кто выиграет при оптимальной игре обоих игроков – игрок, делающий первый ход или игрок, делающий второй ход.

### Формат входных данных

Каждый тест состоит из нескольких наборов входных данных. В первой строке находится одно целое число  $t$  ( $1 \leq t \leq 10^4$ ) — количество наборов входных данных. Далее следует описание наборов входных данных. Каждый набор состоит из трёх строк, каждая из которых содержит информацию о соответствующем стеке. В  $i$ -й ( $1 \leq i \leq 3$ ) строке содержатся два числа  $top_i$  и  $n_i$  ( $top_i = \text{«А»}$  или  $top_i = \text{«В»}$ ,  $1 \leq n_i \leq 10^9$ ), обозначающие символ на вершине  $i$ -го стека и количество элементов в нём. Так как символы чередуются, то такая информация однозначно задаёт стек.

### Формат выходных данных

Для каждого набора входных данных выведите 1, если выигрывает первый игрок для этого набора, и 0 иначе.

### Пример

стандартный ввод	стандартный вывод
5	1
A 1	1
A 1	1
A 1	0
A 1	0
A 2	
A 3	
A 5	
B 90	
A 11	
B 10	
B 10	
B 100	
A 155	
B 155	
A 15	

## Задача С. Вот такая формула

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 4 секунды  
Ограничение по памяти: 1024 мегабайта

Обозначим за  $S(x)$  сумму цифр целого положительного числа  $x$ . Пусть  $S^0(x) = x$ ,  $S^1(x) = S(x)$ ,  $\dots$ ,  $S^k(x) = S(S^{k-1}(x))$ . Вам даны числа  $n$  и  $k$ . Найдите все решения уравнения  $x + S(x) + S^2(x) + S^3(x) + \dots + S^k(x) = n$ .

### Формат входных данных

В первой строке вам дано количество наборов тестовых данных  $1 \leq T \leq 500$ . В следующих  $T$  строках заданы тесты – каждый из них описывается парой  $1 \leq n \leq 10^{12}$ ,  $0 \leq k \leq 10^{10}$ .

### Формат выходных данных

Для каждого теста в отдельной строке выведите все решения уравнения через пробел в порядке возрастания. Вывод начинайте с числа решений, далее выводите все решения. Выведенные решения для каждого теста должны быть различными.

### Пример

стандартный ввод	стандартный вывод
3	2
33 2	17 23
1654 4	1
992 3	1625
	1
	959

## Задача D. Групповые логические операции

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 1024 мегабайта

В современных системах облачной инфраструктуры, в частности, в MWS CDN, важное значение имеет быстроедействие на уровне базовых запросов.

Вас пригласили на стажировку в МТС и поручили решить следующую задачу, связанную с низкоуровневой оптимизацией архитектуры системы.

У вас есть набор из  $n$  нулей и  $m$  единиц, а также набор из  $a$  логических операций  $OR$  и набор из  $n + m - 1 - a$  логических операций  $AND$ . Сначала вы выписываете все нули и единицы, которые у вас есть, в строку, а затем над каждой парой соседних элементов в строке пишете одну из логических операций, которые у вас есть. Вы должны использовать ровно столько операций каждого типа, сколько у вас имеется в наличии. После этого вы считаете результат полученного выражения следующим образом: к каждой паре соседних элементов в строке применяется операция, записанная над ней, и результат этой операции прибавляется к общему результату. Результаты операций считаются, как результаты классических логических операций «И» и «ИЛИ», а именно:

- $0 OR 0 = 0$
- $0 OR 1 = 1$
- $1 OR 0 = 1$
- $1 OR 1 = 1$
- $0 AND 0 = 0$
- $0 AND 1 = 0$
- $1 AND 0 = 0$
- $1 AND 1 = 1$

Ваша задача – максимизировать итоговый результат, если вы оптимальным образом расставите числа и операции.

### Формат входных данных

В первой строке записано число  $1 \leq T \leq 10^4$  – количество наборов тестовых данных. В следующих  $T$  строках содержатся независимые запросы – в каждой строке вам даны три числа  $0 \leq n \leq 10^9$ ,  $0 \leq m \leq 10^9$  и  $0 \leq a \leq n + m - 1$ . Гарантируется, что  $n + m > 0$ .

### Формат выходных данных

Для каждого запроса выведите ответ в отдельной строке – максимальный результат, если упорядочить числа и операции оптимальным образом.

### Пример

стандартный ввод	стандартный вывод
2	8
1 8 7	0
100 0 0	

## Задача Е. Дивизионом ниже

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	1024 мегабайта

Организаторам фестиваля РуКод на площадках в день соревнования приходится проверять длинные списки, сверяя, прошёл ли участник отбор и в каком дивизионе он участвует.

Вы решили разработать систему, которая бы сразу позволяла распознать  $\text{div EF}$ . Так как на участниках попрактиковаться удастся только в следующем сезоне, вы решили попрактиковаться на числах. Ваша задача — проверить число на соответствие  $\text{div EF}$ . Число соответствует  $\text{div EF}$ , если оно делится на число, которое в некоторой системе счисления соответствует записи EF (считается, что в системах счисления используются сначала цифры 0-9, потом заглавные буквы от A до Z; для всех других цифр, меньших основания, какие-то знаки тоже существуют, но в рамках этой задачи не важны).

Например, 42 не является  $\text{div EF}$ -числом, а 478 — является (так как оно делится на  $239 = EF_{16}$ ).

В случае, если число является  $\text{div EF}$ -числом, найдите минимальное основание системы счисления, в которой один из делителей этого числа имеет вид EF.

### Формат входных данных

Первая строка входных данных содержит одно целое число  $n$  ( $1 \leq n \leq 10^{12}$ ).

### Формат выходных данных

Если заданное число  $n$  является  $\text{divEF}$ -числом, выведите минимальное основание системы счисления, в которой один из делителей этого числа имеет вид EF. В противном случае выведите  $-1$ .

### Примеры

стандартный ввод	стандартный вывод
42	-1
478	16

## Задача F. Египтологические треугольники

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 1024 мегабайта

Назовём *египтологическими* прямоугольные треугольники со следующим свойством: все стороны треугольника имеют целые длины и периметр треугольника делится хотя бы на один из его катетов. Например, классический египетский треугольник имеет периметр  $3+4+5=12$  и является египтологическим: его периметр делится на 3.

Ваша задача — по заданному  $n$  выяснить, сколько существует различных египтологических треугольников, у которых один из катетов имеет длину  $n$ .

### Формат входных данных

Входные данные содержат одно целое число  $n$  ( $1 \leq n \leq 10^{12}$ ).

### Формат выходных данных

Выведите одно целое число — количество различных египтологических треугольников, у которых один из катетов имеет длину  $n$ .

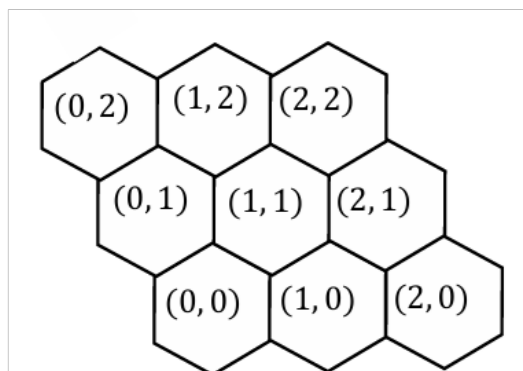
### Пример

стандартный ввод	стандартный вывод
6	1

## Задача G. Ёмкости с бензином

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 1024 мегабайта

В компьютерной игре «Mighty Mages and Heroes» игровое поле состоит из  $(w + 1) \times (h + 1)$  шестиугольных клеток. Вы играете за дракона, ваш дракон находится на поле  $(0, 0)$ , его пещера — на поле  $(w, h)$ . На рисунке ниже приведён пример нумерации нижнего левого угла поля (окрестностей клетки  $(0, 0)$ ).



Если дракон находится в клетке  $(x, y)$ , то он может пойти на одну из клеток  $(x - 1, y - 1)$ ,  $(x - 1, y)$ ,  $(x, y - 1)$ ,  $(x + 1, y)$ ,  $(x, y + 1)$ ,  $(x + 1, y + 1)$ , при этом клеток с  $x < 0$ ,  $x > w$ ,  $y < 0$  и  $y > h$  не существует (и дракон не может пойти на такую клетку).

При этом на клетки расставлены ёмкости с бензином. Изначально пожароопасность всех клеток равна 0. Если ёмкость с бензином объёмом в  $a$  литров ставится на клетку  $(x, y)$ , то пожароопасность меняется следующим образом.

Пусть  $j$  — минимальное количество пересечений границ клеток при перемещении от клетки  $(x, y)$  до клетки  $(u, v)$ . Тогда, если  $a - j$  положительно, степень пожароопасности клетки  $(u, v)$  увеличивается на  $a - j$ . Если  $a - j$  меньше или равно 0, то степень пожароопасности клетки  $(u, v)$  не изменяется.

Всего было расставлено  $n$  ёмкостей с бензином одна за другой,  $i$ -я ёмкость ставится на клетку  $(x_i, y_i)$  и будет иметь объём  $a_i$  литров.

Вы хотите провести дракона по такому маршруту, чтобы максимальная степень  $f_{max}$  пожароопасности среди всех клеток, через которые он проходит, была минимально возможной.

### Формат входных данных

Первая строка входных данных содержит два целых числа  $w$  и  $h$  — размеры поля ( $1 \leq h, w \leq 2 \cdot 10^5$  и  $h \times w \leq 2 \cdot 10^5$ ) и количество ёмкостей с бензином  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ).

$i$ -я из последующих  $n$  строк задаёт координаты  $i$ -й ёмкости  $x_i$  ( $0 \leq x_i \leq w$ ),  $y_i$  ( $0 \leq y_i \leq h$ ) и её объём  $a_i$  ( $1 \leq a_i \leq 10^9$ ) для  $i$ -го распыления.

### Формат выходных данных

Выведите одно число — максимальное значение степени пожароопасности  $f_{max}$ .

### Пример

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

## Задача Н. Жизнь лягушек

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	1024 мегабайта

На болоте выстроились в круг  $n$  кувшинок. Пронумеруем их по часовой стрелке от 1 до  $n$ . Две лягушки — Анфиса и Василиса — сидят на каких-то двух из этих кувшинок, первая лягушка сидит на кувшинке  $a$ , а вторая лягушка — на кувшинке  $b$ . Им нужно встретиться, то есть оказаться на одной кувшинке. Для этого они могут совершать следующие ходы(ход состоит в перемещении на соседнюю кувшинку):

- Анфиса делает ход по часовой стрелке и тратит на это  $t_1$  секунд
- Анфиса делает ход против часовой стрелки и тратит на это  $t_2$  секунд
- Василиса делает ход по часовой стрелке и тратит на это  $t_3$  секунд
- Василиса делает ход против часовой стрелки не тратит на это  $t_4$  секунд
- Обе лягушки делают ходы навстречу друг другу — Анфиса делает ход по часовой стрелке, а затем Василиса делает ход против часовой стрелки. Такое действие недопустимо, если после хода Анфиса окажется на той же кувшинке, что и Василиса до хода — ведь вторая обязательно должна походить. На это действие тратится суммарно  $t_5$  секунд
- Обе лягушки делают ходы навстречу друг другу — Анфиса делает ход против часовой стрелки, а затем Василиса делает ход по часовой стрелке. Такое действие недопустимо, если после хода Анфиса окажется на той же кувшинке, что и Василиса до хода — ведь вторая обязательно должна походить. На это действие тратится суммарно  $t_6$  секунд

Ваша задача — определить минимальное время в секундах, через которое могут встретиться две лягушки, если они будут действовать оптимально.

### Формат входных данных

В первой строке вам дано количество запросов в одном тесте  $1 \leq T \leq 10^4$ . В следующих  $T$  строках даны сами запросы. Каждый запрос характеризуется девятью числами — количеством кувшинок  $1 \leq n \leq 10^9$ , позициями лягушек  $1 \leq a, b \leq n$ (сначала дана позиция Анфисы, а потом Василисы) и временами  $1 \leq t_1, t_2, t_3, t_4, t_5, t_6 \leq 10^9$ . Каждый запрос представляет собой независимую задачу, которую вы должны решить.

### Формат выходных данных

Для каждого запроса в отдельной строке выведите минимальное время в секундах, которое потребуется на то, чтобы лягушки встретились.

### Пример

стандартный ввод	стандартный вывод
3	5
8 7 2 4 1 2 3 3 2	138
100 15 61 8 11 14 7 6 6	447
448 1 448 1 10000 10000 1 2 1000000	



## Задача I. Задача за 300

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

*Это интерактивная задача*

Екатерина учится в седьмом классе и является энтузиасткой олимпиадного программирования. Так что она не только активно тренируется решать задачи, чтобы стать абсолютной победительницей Всероссийской олимпиады по информатике, она ещё и хорошо знает историю олимпиад, в частности, тот факт, что первая Всесоюзная олимпиада по информатике прошла в 1987-1988 учебном году. Так что Екатерина считает все годы с 1 по 1986 включительно *доисторическими*.

Перед командной тренировкой Екатерина предложила сыграть в следующую игру: она загадывает «доисторический» год (то есть целое число от 1 до 1986 включительно), а остальная команда задаёт запросы.

Каждый запрос представляет собой целое число  $k$ , также находящееся в интервале от 1 до 1986 включительно. В ответ Екатерина вычисляет наибольший общий делитель  $d$  чисел  $n$  и  $k$ , проверяет, записано ли  $d$  на бумажке (перед началом игры бумажка пустая), если записано — сообщает «есть», иначе записывает на бумажку  $d$  и отвечает «нет».

Чтобы выиграть, команда должна угадать число за 300 или менее попыток.

Сокомандники менее опытные в олимпиадном программировании, чем Екатерина, и просят вас помочь решить задачу.

### Протокол взаимодействия

Взаимодействие начинается программа жюри, играющая за Екатерину. Она выводит одно целое число  $t$  — количество сценариев ( $1 \leq t \leq 50$ ).

Далее в каждом сценарии взаимодействие начинается ваша программа. Чтобы задать запрос, выведите знак вопроса (?), а затем, через пробел, выведите число  $k$  ( $1 \leq k \leq 1986$ ). Программа жюри выведет 1, если наибольший общий делитель  $n$  и  $k$  уже встречался ранее в качестве ответа на какой-то из предыдущих запросов, и 0 в противном случае.

Чтобы ответить на запрос, выведите восклицательный знак (!), а затем, через пробел, выведите число  $n$ . При подсчёте количества запросов вывод ответа не учитывается как запрос.

Если вы уложились в требуемое количество запросов и число  $n$  угадано правильно, программа жюри переходит к следующему сценарию. Если же что-то пошло не так, вы получите вердикт тестирующей системы об ошибке.

Гарантируется, что программа жюри не изменяет число  $n$  после начала процесса взаимодействия (то есть интерактор не является адаптивным).

### Пример

стандартный ввод	стандартный вывод
1	? 1
0	? 995
1	? 993
1	? 991
0	? 2
0	! 1982

## Замечание

В примере из условия первым ответом в список занесена единица. Далее наибольший общий делитель для чисел 995 и 993 уже есть в списке (то есть равен 1), наибольший общий делитель для 991 отсутствует в списке, то есть он не равен 1; следовательно, он равен 991 (так как 991 — простое). Чисел, кратных 991, всего 2, так что проверяем делимость на 2 и находим ответ 1982.

## Задача J. Интервью с тренером

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	1024 мегабайта

*Это задача с двойным запуском*

...Во время трансляции с соревнований по спортивному программированию гость студии — известный в узких кругах тренер — получил анонимное сообщение: «если хотите узнать сложность задач на предстоящий тур, во время рекламной паузы идите в комнату 322». Речь шла о финальном туре соревнований, в котором планировалось 10 задач. Это могло быть и провокацией — тогда любая попытка связаться с командой приведёт к дисквалификации. Тренер посмотрел на программу беседы — после перерыва ведущий планировал задать вопросы по двум недавним раундам ForceCoders — один раунд был из 7 задач, а второй из 5, а также попросить сделать прогноз, какую из 10 задач на предстоящем финале сдадут самой первой. У тренера появилась мысль, и он связался с командой до того, как пошёл в комнату 322.

...В комнате 322 тренер действительно увидел список 10 задач, отсортированных по сложности. Задачи, как обычно, пронумерованы заглавными латинскими буквами от A до J. Тренер запомнил список и вернулся в студию. И ведущий задал три последних вопроса — попросил упорядочить задачи раундов из семи и из пяти задач по степени оригинальности, а также спросил, какая задача из 10 будет на предстоящем финале сдана первой. Тренер мог выбирать, в каком порядке отвечать на эти вопросы. Ответом на первый и второй должны быть упорядоченные буквы задач, ответом на третий — одна буква от A до J.

Однако бдительные организаторы, догадывавшиеся о цели визита в комнату 322, «запикали» названную тренером букву задачи. Тем не менее, смотрящая трансляцию команда смогла по этому фрагменту интервью восстановить требуемый порядок задач.

...Вам поручено в рамках внутреннего расследования жюри об утечке информации повторить сделанное тренером и командой и передать порядок задач с помощью трёх сообщений — переданных в выбранном вами порядке строки длины 7 с названиями задач от A до G, расставленными в каком-то порядке, аналогичной строки с названиями задач от A до E и одной буквы, которая будет «запикана» на трансляции.

### Формат входных данных

Ваша программа будет запущена два раза, первый раз в режиме **coach**, второй раз в режиме **team**.

Если задача запущена в режиме **coach**, то первой строкой входных данных будет **coach**. Далее следует одно целое число  $t$  ( $1 \leq t \leq 10^5$ ) — количество тестовых примеров. Каждый тестовый пример состоит из одной строки длины 10, составленной из заглавных латинских букв от A до J, причём каждая буква встречается один раз.

Если задача запущена в режиме **team**, то первой строкой входных данных будет **team**. Далее следует одно целое число  $t$  ( $1 \leq t \leq 10^5$ ) — количество тестовых примеров. Каждый тестовый пример располагается на одной строке и состоит из трёх слов через пробел — ответов тренера на вопросы в трансляции. Одна строка содержит 7 заглавных латинских букв от A до G, причём каждая буква встречается ровно один раз, вторая — 5 заглавных латинских букв от A до E, причём каждая буква встречается ровно один раз, третья — букву A (вне зависимости от того, какая буква была названа тренером). Порядок блоков сохраняется.

### Формат выходных данных

Если задача запущена в режиме **coach**, то для каждого тестового примера вы должны вывести три строки. Одна строка из 7 заглавных латинских букв от A до G, причём каждая буква встречается ровно один раз, вторая из 5 заглавных латинских букв от A до E, причём каждая буква встречается ровно один раз, третья — одна буква от A до J. Порядок блоков определяет ваша программа.

Если задача запущена в режиме **team**, выведите одну строку для каждого тестового примера — восстановленный порядок сложности задач в предстоящем туре, то есть строку длины 10, которая

была получена в режиме coach.

## Примеры

стандартный ввод	стандартный вывод
coach 1 ABCDEFGHIJ	J BEDACFG CEBAD
team 1 A BEDACFG CEBAD	ABCDEFGHIJ

## Задача К. Йодосодержащие минералы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

Для презентации крупного проекта по разработке йодосодержащих минералов топ-менеджер Богдан получил последовательность целых неотрицательных чисел  $a_1, a_2, \dots, a_n$ . Некоторые элементы в последовательности равны нулю — это значит, что требуемый параметр ещё не определён. Посоветовавшись с LLM MWS Copilot, Богдан решил, что стоит нули заменить каким-то целым числом  $x$ .

Опытный инвестор сразу замечает потенциальный рост. Поэтому впечатление, которое презентация производит на инвесторов, зависит от длины наибольшей строго возрастающей подпоследовательности. Обозначим зависимость этой длины от значения  $x$  как  $f(x)$ .

Требуется найти  $\sum_{i=1}^n i \cdot f(i)$ .

Напоминаем, что длина наибольшей строго возрастающей подпоследовательности последовательности  $s_1, s_2, \dots, s_m$  — наибольшее  $k$  такое, что существуют  $1 \leq i_1 < i_2 < \dots < i_k \leq m$ , удовлетворяющие  $s_{i_1} < s_{i_2} < \dots < s_{i_k}$ .

### Формат входных данных

Входные данные состоят из нескольких тестовых примеров.

Каждый тестовый пример состоит из двух строк. В первой строке содержится одно целое число  $n$  — длина последовательности  $a$  ( $1 \leq n \leq 10^5$ ). Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  — элементы последовательности ( $0 \leq a_i \leq n$ ).

Гарантируется, что сумма  $n$  по всем тестовым примерам не превосходит  $2.5 \cdot 10^5$ .

Входные данные завершаются тестовым примером с  $n = 0$ , обрабатывать который не следует.

### Формат выходных данных

Для каждого тестового примера в отдельной строке выведите одно целое число — ответ к задаче.

### Пример

стандартный ввод	стандартный вывод
2	3
1 1	24
4	56
2 0 3 0	3
6	
3 0 6 1 0 4	
2	
0 0	
0	

## Задача L. Крош, массив и запросы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	1024 мегабайта

У Кроша есть массив из  $n$  неотрицательных чисел  $a_1, a_2, \dots, a_n$ . Обозначим за  $f(x)$  минимальное число вида  $2^k$ ,  $k \geq 0$ , которое строго больше  $x$ .

Крош может сделать с массивом некоторые операции. За одну операцию он может выбрать некоторое  $1 \leq i \leq n$  и заменить  $a_i$  на  $f(a_i) - 1 - a_i$ . Есть дополнительное ограничение: для каждого индекса  $i$  он может применить операцию к числу с этим индексом не более одного раза.

Крош хочет, чтобы побитовое исключающее или полученных элементов массива было как можно больше. Помогите ему понять, какого максимального его значения можно достичь.

Ответьте на  $q$  запросов. В  $i$ -м запросе вам даются два числа  $l_i$  и  $r_i$ , и вы независимо решаете задачу для подмассива исходного массива, где элементы берутся, начиная от  $l_i$  и заканчивая  $r_i$ .

### Формат входных данных

В первой строке вам дано число  $1 \leq n \leq 2 \cdot 10^5$ , обозначающее количество чисел в массиве у Кроша, и число  $1 \leq q \leq 2 \cdot 10^5$ , обозначающее количество запросов. В следующей строке вам даны элементы массива – неотрицательные целые числа  $0 \leq a_i \leq 10^9$ . В следующих  $q$  строках содержатся запросы –  $i$ -й запрос описывается двумя числами  $1 \leq l_i \leq r_i \leq n$ .

### Формат выходных данных

Для каждого запроса выведите ответ на него в отдельной строке.

### Пример

стандартный ввод	стандартный вывод
8 5	7
3 5 7 6 2 4 3 2	4
1 8	7
2 4	7
2 8	6
3 7	
4 8	

## Задача М. Люди, не расходитесь!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

После завершения тура чемпионата РуКод участники всех дивизионов на одной из площадок выстроились для фотографирования в ряд. Как известно, каждый участник относит себя к одному из шести дивизионов от А до F.

По задумке местных организаторов, каждый участник держит в руках табличку с буквой своего дивизиона. «Минуточку!» — сказал фотограф...

Наблюдая за процессом со стороны, тренер одной из команд задумался: если представить, что буквы на участниках слева направо — шестнадцатеричное число, задающее время в секундах, то какое минимальное количество участников должно подойти, чтобы время стало задавать целое количество минут, и из каких дивизионов будут эти участники? Участники могут подходить как справа, так и слева, но не могут становиться между какими-то другими участниками.

### Формат входных данных

Первая строка входных данных состоит из букв от А до F. Строка непуста, её длина не превосходит  $10^5$ .

### Формат выходных данных

Если строка, интерпретированная как шестнадцатеричное число, и так делится на 60, выведите ОК. Иначе выведите строку, составленную из минимального количества букв дивизионов, задающую список участников, которые могут добавиться к расстановке с её концов так, чтобы число стало делиться на 60. Если таких строк несколько, выведите лексикографически наименьшую из них. При этом порядок букв в строке не обязан совпадать с порядком, в котором участники будут добавляться к расстановке.

### Примеры

стандартный ввод	стандартный вывод
ABCD	AACC
ABEDC	OK