

Задача 1. ННОИ

<i>Ввод</i>	<code>nnoi.in</code> или стандартный ввод
<i>Вывод</i>	<code>nnoi.out</code> или стандартный вывод
<i>Ограничение по времени</i>	1 секунда
<i>Ограничение по памяти</i>	256 мегабайт
<i>Максимальный балл за задачу</i>	200

В 2018 году проводится четырнадцатая Нижегородская городская олимпиада школьников по информатике имени В. Д. Лелюха. Считая, что далее каждый год будет проводиться одна олимпиада, определите, в каком году будет проводиться N -я?

Формат входных данных

Вводится одно целое число N — номер олимпиады ($14 \leq N \leq 1000$).

Формат выходных данных

Выведите одно число — год, в котором будет проводиться N -я олимпиада.

Пример

<i>Ввод</i>	<i>Вывод</i>
34	2038
14	2018

Задача 2. Удобно для всех

<i>Ввод</i>	contest.in или стандартный ввод
<i>Вывод</i>	contest.out или стандартный вывод
<i>Ограничение по времени</i>	1 секунда
<i>Ограничение по памяти</i>	256 мегабайт
<i>Максимальный балл за задачу</i>	100

В далеком будущем на Земле сутки делятся n часов, и, соответственно, есть n часовых поясов. Местное время в соседних часовых поясах различается на час. При записи местного времени для числа часов используются числа от 1 до n , т.е. времени «0 часов» не бывает, вместо этого бывает « n часов». Когда местное время в 1-м часовом поясе 1 час, местное время в i -м часовом поясе i часов.

Некоторая платформа по проведению онлайн-соревнований по программированию хочет провести соревнование длиной в один час, причем так, чтобы начало соревнования совпало бы с началом какого-то часа (во всех часовых зонах). Платформа знает, что из i -го часового пояса в соревновании хотят принять участие a_i человек. Каждый человек примет участие, если соревнование начнется не раньше s часов 00 минут местного времени (по часовому поясу этого человека), а закончится — не позже f часов 00 минут местного времени: в это время человек на работе, а, как известно, на работе не жалко потратить час на самообразование. Начало и конец рабочего дня s и f одинаковы для всех часовых поясов. После f часов 00 минут люди уже не работают, т.е. если соревнование начинается в f часов 00 минут местного времени, то люди участвовать в нем не будут.

Помогите онлайн-платформе выбрать такой час, когда наибольшее число людей примут участие в соревновании.

Формат входных данных

В первой строке записано одно целое число n — длина суток в часах ($2 \leq n \leq 100\,000$).

Во второй строке через пробел записаны n натуральных чисел a_1, a_2, \dots, a_n — количество людей в i -м часовом поясе, которые хотят принять участие в соревновании ($1 \leq a_i \leq 10\,000$).

В третьей строке записаны через пробел два натуральных числа s и f — времена начала рабочего дня и конца рабочего дня по местному времени, соответственно ($1 \leq s < f \leq n$).

Формат выходных данных

В единственной строке выведите одно целое число — время начала соревнований по местному времени первого часового пояса, такое, чтобы для максимального числа людей соревнование прошло в течение рабочего дня. Если ответов несколько, выведите минимальный.

Система оценивания

В тестах общей стоимостью 30 баллов N не будет превосходить 10 000.

Пример

<i>Ввод</i>	<i>Вывод</i>
3 1 2 3 1 3	3
5 1 2 3 4 1 1 3	4

Примечание: В первом примере оптимально начать соревнование в 3 часа по времени первого часового пояса. Тогда во втором часовом поясе будет 1 час, а в третьем — 2 часа. Только один человек из первого часового пояса не примет участия в соревновании, т. к. он заканчивает работать в момент начала соревнования.

Во втором примере в соревновании примут участие люди из третьего и четвертого часовых поясов.

Задача 3. Честный дележ

<i>Ввод</i>	<code>divide.in</code> или стандартный ввод
<i>Вывод</i>	<code>divide.out</code> или стандартный вывод
<i>Ограничение по времени</i>	1 секунда
<i>Ограничение по памяти</i>	256 мегабайт
<i>Максимальный балл за задачу</i>	100

Трое друзей отправились на рыбалку. Они беззаботно ловили рыбу, наслаждаясь природой и хорошей компанией.

Но вот подошла к концу совместная поездка, и стало понятно, что кто-то поймал больше, а кто-то меньше. Один из друзей, поймавший меньше всех, неожиданно заявил, что если он приедет домой с такими результатами, то его жена, объявив его жалким неудачником, уйдет от него. Поэтому было решено разделить весь улов на троих. Вам надо придумать, как организовать обмен рыбы между друзьями, чтобы все в итоге вернулись домой с одинаковым уловом.

Более формально, вам дан список из n попыток друзей поймать рыбу. Для каждой попытки известно, кто пытался поймать рыбу, и сколько он рыб он вытащил (в штуках). Вам требуется вывести таблицу размера 3 на 3, где на j -м месте в i -й строке будет находиться число $a_{i,j}$, указывающее, сколько рыб i -й человек должен получить от j -го. Это число может быть как положительным, так и отрицательным (в случае, если i -й человек должен отдать рыбу j -му). Для вашей таблицы должно выполняться равенство $a_{i,j} = -a_{j,i}$ (в частности, из этого следует, что $a_{i,i} = 0$).

Суммарный итоговый улов каждого из друзей должен быть одинаковыми (улов i -го человека складываются из той рыбы, которую он поймал сам, и той рыбы, которую он получит в результате итогового обмена).

Гарантируется, что существует такой способ обменяться рыбой, чтобы каждый рыбак ушел домой с одинаковым целым количеством рыб.

Формат входных данных

В первой строке вводится одна строка, состоящая только из букв латинского алфавита — имя первого из друзей. Имя непустое и его длина не превосходит 10. Во второй и третьей строках в таком же формате вводятся имена второго и третьего друга. Гарантируется, что все имена различны.

В четвертой строке вводится одно целое число n — количество попыток поймать рыбу ($1 \leq n \leq 100$).

В следующих n строках вводятся сами попытки — сначала имя человека, который удил рыбу (гарантируется, что это имя было указано в одной из первых трех строк входных данных), затем через пробел одно целое число c_i — сколько рыб он поймал ($1 \leq c_i \leq 10^4$).

Гарантируется, что суммарный улов можно разделить на троих поровну, оставляя рыбы целыми.

Формат выходных данных

Выведите 3 строки, в каждой по 3 целых числа. В i -й строке j -е число должно быть равно $a_{i,j}$ — сколько рыб (в штуках) должен получить i -й человек от j -го (нумерация друзей соответствует порядку, в котором они даны во входных данных).

Для выведенных чисел должны выполняться следующие условия:

- $a_{i,j} = -a_{j,i}$;
- $a_{i,j}$ — целое число, удовлетворяющее ограничению $|a_{i,j}| \leq 10^7$;
- суммарный улов каждого из друзей (с учетом попыток, указанных во входных данных) одинаков.

Можно показать, что хотя бы один способ подобрать такие $a_{i,j}$ существует.

Пример

<i>Ввод</i>	<i>Вывод</i>
Vlad	0 0 1
Kolya	0 0 -2
Misha	-1 2 0
1	
Kolya 3	

Примечание: В первом примере ловил рыбу только Коля. Один из вариантов честного дележа такой: Коля отдает Мише 2 рыбы, потом Миша отдает Владу 1 рыбу.

Задача 4. Бутявочная ферма

<i>Ввод</i>	butyavki.in или стандартный ввод
<i>Вывод</i>	butyavki.out или стандартный вывод
<i>Ограничение по времени</i>	1 секунда
<i>Ограничение по памяти</i>	256 мегабайт
<i>Максимальный балл за задачу</i>	100

У мальчика Димы есть своя бутявочная ферма. Скоро на ней вырастут N бутявок, и Диме надо будет отвезти их на рынок продавать.

Бутявок надо перевозить в стеклянных банках. Если какая-то банка будет заполнена не полностью, то бутявки в этой банке закузываются, поэтому каждую банку надо заполнять полностью.

Дима может покупать банки на стеклозаводе. Завод производит K типов банок, банки i -го типа вмещают a_i бутявок каждая. Дима может заказать на заводе сколько угодно банок, но, чтобы получить оптовую скидку, он должен заказывать банки только какого-то одного типа.

Естественно, Дима хочет заказать банки так, чтобы можно было каждую банку заполнить бутявками доверху и отвезти на рынок; если при этом какие-то бутявки не влезут, то Дима оставит этих бутявок на ферме.

Определите, сколько банок и какого типа должен заказать Дима, чтобы увезти на рынок как можно больше бутявок.

Формат входных данных

Первая строка входных данных содержит два целых числа N и K — количество бутявок, которые вырастут у Димы на ферме, и количество типов банок, которые производит стеклозавод ($0 \leq N \leq 10^{18}$, $1 \leq K \leq 10^5$).

Во второй строке находятся K целых чисел a_1, a_2, \dots, a_K — вместимости каждого типа банок ($1 \leq a_i \leq 10^{18}$ для всех i).

Формат выходных данных

Выведите два числа — номер типа банок, которые должен заказать Дима, и их количество. Типы банок нумеруются от 1 до K в том порядке, как они описаны во входных данных.

Если верных ответов несколько, вы можете вывести любой из них.

Система оценивания

В тестах общей стоимостью 25 баллов все числа во входных данных не будут превосходить 1000.

В тестах общей стоимостью 50 баллов все числа во входных данных не будут превосходить 10^5 .

Пример

<i>Ввод</i>	<i>Вывод</i>
19 3 5 4 10	2 4
28 3 5 6 30	1 5

Задача 5. Максимизируй!

<i>Ввод</i>	maximum.in или стандартный ввод
<i>Вывод</i>	maximum.out или стандартный вывод
<i>Ограничение по времени</i>	1 секунда
<i>Ограничение по памяти</i>	256 мегабайт
<i>Максимальный балл за задачу</i>	100

Дано множество S из натуральных чисел, изначально пустое. С ним производятся операции двух типов:

1. Добавить в S натуральное число, не меньшее всех уже лежащих в множестве элементов.
2. Найти непустое подмножество s множества S такое, что значение $\max(s) - \text{mean}(s)$ максимально. Здесь $\max(s)$ обозначает максимум среди всех элементов s , $\text{mean}(s)$ — среднее арифметическое всех элементов s . Вывести величину $\max(s) - \text{mean}(s)$.

Формат входных данных

В первой строке задано единственное натуральное число Q ($1 \leq Q \leq 5 \cdot 10^5$) — количество операций. В каждой из следующих Q строк следует описание очередной операции. Для операций типа 1 в строке будут записаны два числа вида 1 и x , где x ($1 \leq x \leq 10^9$) — натуральное число, которое нужно добавить в S . Гарантируется, что x будет не меньше любого элемента S на данный момент. Для операции типа 2 в соответствующей строке будет записано ровно одно число 2.

Гарантируется, что первая операция будет иметь тип 1, то есть множество S не будет пустым при поступлении операций типа 2.

Формат выходных данных

Выведите ответ на каждую операцию второго типа в том порядке, в котором эти операции шли во входных данных. Каждое число должно располагаться на отдельной строке и содержать минимум шесть верных значащих цифр.

Система оценивания

Решения, правильно работающие для $Q \leq 1400$, будут набирать не менее 30 баллов.

Решения, правильно работающие для $Q \leq 20\,000$, будут набирать не менее 44 баллов.

Пример

<i>Ввод</i>	<i>Вывод</i>
6	0.0000000000
1 3	0.5000000000
2	3.0000000000
1 4	
2	
1 8	
2	
4	2.0000000000
1 1	
1 4	
1 5	
2	

Задача 6. Котлета

Ввод	cutlet.in или стандартный ввод
Вывод	cutlet.out или стандартный вывод
Ограничение по времени	2 секунды
Ограничение по памяти	256 мегабайт
Максимальный балл за задачу	100

Аркадий хочет обедать. Он только что вернулся из магазина, где приобрел полуфабрикат — котлету, которую осталось только поджарить. Надпись на упаковке гласит, что котлету нужно жарить на сковороде на умеренном огне ровно $2n$ секунд, причем сначала нужно ровно n секунд жарить котлету на одной стороне, а затем ровно n секунд — на другой. Аркадий уже нашел сковороду и зажег умеренный огонь, но тут осознал, что, возможно, у него не получится перевернуть котлету ровно через n секунд после начала готовки.

Аркадий слишком занят расстановкой по порядку наборов стикеров в своем любимом мессенджере, и может отвлечься на переворот котлеты только в определенные моменты времени. А именно, есть k промежутков времени, в которые он может это сделать, i -й из них — это отрезок времени с l_i секунд от начала готовки до r_i секунд от начала готовки, включительно. Аркадий решил, что не обязательно переворачивать котлету ровно в середине готовки, вместо этого, он перевернет ее несколько раз таким образом, чтобы суммарно котлета провела n секунд на одной стороне и n секунд на другой.

Помогите Аркадию — узнайте, может ли он выполнить свой план, если он может переворачивать котлету только в указанные отрезки времени, и если да, то какое минимальное число раз ему придется перевернуть котлету.

Формат входных данных

Первая строка содержит два целых числа n и k ($1 \leq n \leq 500\,000$, $1 \leq k \leq 100$) — число секунд, которое котлета должна жариться с каждой из сторон, и число промежутков времени, когда Аркадий может ее переворачивать.

Следующие k строк содержат описания этих промежутков. Каждая строка содержит два целых числа l_i и r_i ($0 \leq l_i \leq r_i \leq 2 \cdot n$), что означает, что Аркадий может перевернуть котлету в любой момент времени, начиная с l_i секунд от начала готовки и заканчивая r_i секундами от начала готовки, включительно. В частности, если $l_i = r_i$, то Аркадий может перевернуть котлету в момент времени $l_i = r_i$. Гарантируется, что $l_i > r_{i-1}$ для всех $2 \leq i \leq k$.

Формат выходных данных

Выведите единственное слово «Hungry», если Аркадий не может пожарить котлету ровно n секунд на одной стороне и ровно n секунд на другой.

В противном случае, выведите в первую строку одно слово «Full», а во вторую — минимальное количество раз, которое ему необходимо перевернуть котлету.

Система оценивания

Тесты, в которых в правильном ответе котлету нужно переворачивать не более двух раз, или Аркадий не может пожарить котлету требуемым образом, имеют суммарную стоимость не менее 30 баллов.

Тесты, в которых $n \leq 1000$, имеют суммарную стоимость не менее 40 баллов.

Пример

Ввод	Вывод
10 2 3 5 11 13	Full 2
10 3 3 5 9 10 11 13	Full 1
20 1 3 19	Hungry

Примечание: В первом примере котлету нужно перевернуть в моменты времени 3 секунды после начала и 13 секунд после начала.

Во втором примере котлету можно перевернуть, как и написано на упаковке, через 10 секунд после начала.

Задача 7. Спасение любви

<i>Ввод</i>	<code>tshirt.in</code> или стандартный ввод
<i>Вывод</i>	<code>tshirt.out</code> или стандартный вывод
<i>Ограничение по времени</i>	1 секунда
<i>Ограничение по памяти</i>	256 мегабайт
<i>Максимальный балл за задачу</i>	100

Валя и Толя — идеальная пара, но даже у них бывают ссоры. Недавно Валя обиделась на своего кавалера, так как он пришел к ней в футболке, надпись на которой отличается от надписи на ее свитере. Теперь она не хочет с ним видаться, а Толя сидит целыми днями в своей комнате и плачет над ее фотографиями.

Эта история так и осталась бы такой печальной, если бы в нее не вмешалась добрая швея-волшебница (бабушка Толи). Ее сердце разрывается, когда она видит молодых людей в ссоре, и поэтому она срочно хочет все исправить. Бабушка уже тайно забрала Валин свитер и Толину футболку, осталось только сделать надписи на них одинаковыми. Для этого она может за одну единицу маны купить заклинание, которое позволяет менять некоторые буквы на одежде. Ваша задача — рассчитать, какое минимальное количество маны придется потратить бабушке Толи для спасения любви молодых людей.

Более формально, надписи на Валином свитере и Толиной футболке — это две строки одинаковой длины n , состоящие только из маленьких букв латинского алфавита. За одну единицу маны бабушка может купить заклинание вида (c_1, c_2) (где c_1 и c_2 — произвольные строчные буквы латинского алфавита), с помощью которого она может сколько угодно раз менять букву c_1 на c_2 (и наоборот) как на Валином свитере, так и на Толиной футболке. Вам требуется найти минимальное количество маны, позволяющее приобрести набор заклинаний, с помощью которого можно сделать надписи одинаковыми. Также вам необходимо вывести соответствующий набор заклинаний.

Формат входных данных

В первой строке дано единственное натуральное число n — длина надписей ($1 \leq n \leq 10^5$).

Во второй строке дана строка длины n , состоящая только из маленьких букв латинского алфавита — надпись на Валином свитере.

В третьей строке в таком же формате задана надпись на Толиной футболке.

Формат выходных данных

В первой строке выведите одно целое число — минимальное количество маны t для спасения любви молодых.

В следующих t строках выведите по две строчные буквы латинского алфавита через пробел — заклинания, которые нужно приобрести бабушке Толи. Заклинания и буквы в заклинаниях можно выводить в любом порядке.

Если оптимальных ответов несколько, выведите любой.

Пример

<i>Ввод</i>	<i>Вывод</i>
3 abb dad	2 a d b a
8 drpepper cocalola	7 l e e d d c c p p o o r r a

Примечание: В первом примере достаточно купить два заклинания: («a», «d») и («b», «a»). Тогда первые буквы совпадут, когда мы поменяем букву «a» на «d». Вторые совпадут, когда поменяем «b» на «a». Третьи совпадут, когда поменяем сначала «b» на «a», потом «a» на «d».