

Задача А. Любимые числа Поликарпа

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Поликарп мечтает стать программистом и фанатеет от степеней двойки. Среди двух чисел ему больше нравится то, которое делится на большую степень числа 2.

По заданной последовательности целых положительных чисел $a_1, a_2, \dots, a_n$ требуется найти r — максимальную степень числа 2, на которую делится хотя бы одно из чисел последовательности. Кроме того, требуется вывести количество чисел a_i , которые делятся на r .

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 100$) — длина последовательности a .

Во второй строке записана последовательность целых чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите два числа:

- r — максимальную степень двойки, на которую делится хотя бы одно из чисел заданной последовательности,
- количество элементов последовательности, которые делятся на r .

Примеры

стандартный ввод	стандартный вывод
5 80 7 16 4 48	16 3
4 21 5 3 33	1 4

Замечание

В первом тестовом примере максимальная степень двойки, на которую делится хотя бы одно число, равна $16 = 2^4$, на неё делятся числа 80, 16 и 48.

Во втором тестовом примере все четыре числа нечётные, поэтому делятся только на $1 = 2^0$. Это и будет максимальной степенью двойки для данного примера.

Задача В. Этажи

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Есть n -подъездный дом, в каждом подъезде по m этажей, и на каждом этаже каждого подъезда ровно k квартир. Таким образом, в доме всего $n \cdot m \cdot k$ квартир. Они пронумерованы естественным образом от 1 до $n \cdot m \cdot k$, то есть первая квартира на первом этаже в первом подъезде имеет номер 1, первая квартира на втором этаже первого подъезда имеет номер $k + 1$ и так далее. Особенность этого дома состоит в том, что он круглый. То есть если обходить его по часовой стрелке, то после подъезда номер 1 следует подъезд номер 2, затем подъезд номер 3 и так далее до подъезда номер n . После подъезда номер n снова идёт подъезд номер 1.

Эдвард живёт в квартире номер a , а Наташа — в квартире номер b . Переход на 1 этаж вверх или вниз по лестнице занимает 5 секунд, переход от двери подъезда к двери соседнего подъезда — 15 секунд, а переход в пределах одного этажа одного подъезда происходит мгновенно. Также в каждом подъезде дома есть лифт. Он устроен следующим образом: он всегда приезжает ровно через 10 секунд после вызова, а чтобы переместить пассажира на один этаж вверх или вниз, лифт тратит ровно 1 секунду. Посадка и высадка происходят мгновенно.

Помогите Эдварду найти минимальное время, за которое он сможет добраться до квартиры Наташи. Считайте, что Эдвард может выйти из подъезда только с первого этажа соответствующего подъезда (это происходит мгновенно). Если Эдвард стоит перед дверью какого-то подъезда, он может зайти в него и сразу окажется на первом этаже этого подъезда (это также происходит мгновенно). Эдвард может выбирать, в каком направлении идти вокруг дома.

Формат входных данных

В первой строке входных данных следуют три числа n, m, k ($1 \leq n, m, k \leq 1000$) — количество подъездов в доме, количество этажей в каждом подъезде и количество квартир на каждом этаже каждого подъезда соответственно.

Во второй строке входных данных записаны два числа a и b ($1 \leq a, b \leq n \cdot m \cdot k$) — номера квартир, в которых живут Эдвард и Наташа, соответственно. Гарантируется, что эти номера различны.

Формат выходных данных

Выведите единственное целое число — минимальное время (в секундах), за которое Эдвард сможет добраться от своей квартиры до квартиры Наташи.

Примеры

стандартный ввод	стандартный вывод
4 10 5 200 6	39
3 1 5 7 2	15

Замечание

В первом тестовом примере Эдвард находится в 4 подъезде на 10 этаже, а Наташа находится в 1 подъезде на 2 этаже. Поэтому Эдварду выгодно сначала спуститься на лифте на первый этаж (на это он потратит 19 секунд, из которых 10 — на ожидание и 9 — на поездку на лифте), затем обойти дом против часовой стрелки до подъезда номер 1 (на это он потратит 15 секунд), и наконец подняться по лестнице на этаж номер 2 (на это он потратит 5 секунд). Таким образом, ответ равен $19 + 15 + 5 = 39$.

Во втором тестовом примере Эдвард живёт в подъезде 2 на этаже 1, а Наташа находится в подъезде 1 на этаже 1. Поэтому Эдварду выгодно просто обойти дом по часовой стрелке до подъезда 1, на это он потратит 15 секунд.

Задача С. Печать условий

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

На тренировку по подготовке к соревнованиям по программированию пришли n команд. Тренер для каждой команды подобрал тренировку, комплект задач для i -й команды занимает a_i страниц. В распоряжении тренера есть x листов бумаги, у которых обе стороны чистые, и y листов, у которых только одна сторона чистая. При печати условия на листе первого типа можно напечатать две страницы из условий задач, а при печати на листе второго типа — только одну. Конечно, на листе нельзя печатать условия из двух разных комплектов задач. Обратите внимание, что при использовании листов, у которых обе стороны чистые, не обязательно печатать условие на обеих сторонах, одна из них может остаться чистой.

Вам предстоит определить максимальное количество команд, которым тренер сможет напечатать комплекты задач целиком.

Формат входных данных

В первой строке входных данных следуют три целых числа n , x и y ($1 \leq n \leq 200\,000$, $0 \leq x, y \leq 10^9$) — количество команд, количество листов бумаги с двумя чистыми сторонами и количество листов бумаги с одной чистой стороной.

Во второй строке входных данных следует последовательность из n целых чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10\,000$), где i -е число равно количеству страниц в комплекте задач для i -й команды.

Формат выходных данных

Выведите единственное целое число — максимальное количество команд, которым тренер сможет напечатать комплекты задач целиком.

Примеры

стандартный ввод	стандартный вывод
2 3 5 4 6	2
2 3 5 4 7	2
6 3 5 12 11 12 11 12 11	1

Замечание

В первом тестовом примере можно напечатать оба комплекта задач. Один из возможных ответов — напечатать весь первый комплект задач на листах с одной чистой стороной (после этого останется 3 листа с двумя чистыми сторонами и 1 лист с одной чистой стороной), а второй комплект напечатать на трех листах с двумя чистыми сторонами.

Во втором тестовом примере можно напечатать оба комплекта задач. Один из возможных ответов — напечатать первый комплект задач на двух листах с двумя чистыми сторонами (после этого останется 1 лист с двумя чистыми сторонами и 5 листов с одной чистой стороной), а второй комплект напечатать на одном листе с двумя чистыми сторонами и на пяти листах с одной чистой стороной. Таким образом, тренер использует все листы для печати.

В третьем тестовом примере можно напечатать только один комплект задач (любой из трёх 11-страничных). Для печати 11-страничного комплекта задач будет израсходована вся бумага.

Задача D. Дефрагментация памяти

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Память компьютера состоит из n ячеек, которые выстроены в ряд. Пронумеруем ячейки от 1 до n слева направо. Про каждую ячейку известно, свободна она или принадлежит какому-либо процессу (в таком случае известен процесс, которому она принадлежит).

Для каждого процесса известно, что принадлежащие ему ячейки занимают в памяти непрерывный участок. С помощью операций вида «переписать данные из занятой ячейки в свободную, а занятую теперь считать свободной» требуется расположить все принадлежащие процессам ячейки в начале памяти компьютера. Другими словами, любая свободная ячейка должна располагаться правее (иметь большой номер) любой занятой.

Вам необходимо найти минимальное количество операций переписывания данных из одной ячейки в другую, с помощью которых можно достичь описанных условий. Допустимо, что относительный порядок ячеек в памяти для каждого из процессов изменится после дефрагментации, но относительный порядок самих процессов должен остаться без изменений. Это значит, что если все ячейки, принадлежащие процессу i , находились в памяти раньше всех ячеек процесса j , то и после перемещений это условие должно выполняться.

Считайте, что номера всех процессов уникальны, хотя бы одна ячейка памяти занята каким-либо процессом.

Формат входных данных

В первой строке входных данных записано число n ($1 \leq n \leq 200\,000$) — количество ячеек в памяти компьютера.

Во второй строке входных данных следуют n целых чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq n$), где a_i равно либо 0 (это означает, что i -я ячейка памяти свободна), либо номеру процесса, которому принадлежит i -я ячейка памяти. Гарантируется, что хотя бы одно значение a_i не равно 0.

Процессы пронумерованы целыми числами от 1 до n в произвольном порядке. При этом процессы не обязательно пронумерованы последовательными числами.

Формат выходных данных

Выведите одно целое число — минимальное количество операций, которое нужно сделать для дефрагментации памяти.

Примеры

стандартный ввод	стандартный вывод
4 0 2 2 1	2
8 0 8 8 8 0 4 4 2	4

Замечание

В первом тестовом примере достаточно двух операций:

1. Переписать данные из третьей ячейки в первую. После этого память компьютера примет вид: 2 2 0 1.
2. Переписать данные из четвертой ячейки в третью. После этого память компьютера примет вид: 2 2 1 0.

Задача Е. Автобус

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	8 секунд
Ограничение по памяти:	256 мегабайт

Вдоль дороги стоят n путешественников. Дорога представляет собой прямую, размерами путешественников можно пренебречь, считая их точками.

Водитель автобуса Василий, благодаря мобильному приложению, знает для каждого путешественника точку x_i , в которой тот стоит. Кроме того, он знает расстояние d_i , которое i -й путешественник хочет проехать на автобусе. Таким образом, i -й путешественник планирует выйти из автобуса в точке $x_i + d_i$. Теперь Василий хочет решить, кого из путешественников он подвезёт, а кого оставит пылиться у дороги.

Василий решил, что сегодня он должен хорошо заработать, поэтому решил перевезти в точности a путешественников. В автопарке есть автобусы любых видов. Чем больше мест в автобусе, тем дороже стоит его аренда.

Помогите Василию определить минимальное количество пассажирских мест в автобусе, которых будет достаточно для перевозки ровно a путешественников. Ни в какой момент времени в автобусе не может быть путешественников больше, чем количество пассажирских мест в автобусе. Василий сам может решить какое конкретно подмножество из a путешественников он перевезёт на автобусе.

Считайте, что автобус всегда едет слева направо (от меньших координат к большим) и начинает свой путь левее самого левостоящего путешественника. Если в одной точке какой-то путешественник должен выйти из автобуса, а другой войти, считайте, что сначала произойдет выход одного путешественника из автобуса, а затем другой путешественник сможет зайти в автобус.

Формат входных данных

В первой строке входных данных следует два целых положительных числа n и a ($1 \leq a \leq n \leq 200\,000$) — количество путешественников, стоящих вдоль дороги, и минимальное количество путешественников, которых Василий хочет подвезти.

В каждой из следующих n строк содержится по два целых числа x_i и d_i ($1 \leq x_i, d_i \leq 10^9$) — координата, в которой находится i -й путешественник, а также расстояние, на которое он хочет переместиться на автобусе. Координаты путешественников заданы в произвольном порядке. В одной точке могут находиться несколько путешественников.

Формат выходных данных

Сначала выведите одно целое число — минимальное количество пассажирских мест в автобусе, которых будет достаточно для перевозки хотя бы a путешественников.

Затем выведите a целых чисел — номера путешественников, которых подвезёт Василий. Путешественники нумеруются, начиная с единицы, в том порядке, в котором заданы во входных данных. Номера можно выводить в произвольном порядке. Если ответов несколько, разрешается вывести любой из них.

Примеры

стандартный ввод	стандартный вывод
3 2 8 9 3 5 1 3	1 1 3
5 4 20 40 10 10 15 5 5 15 20 30	2 4 2 5 1

Замечание

В первом тестовом примере достаточно одноместного автобуса. К примеру, Василий может подвезти третьего и первого путешественников, либо второго и первого путешественников.