Задача A. Великое Лайнландское переселение

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Лайнландия представляет из себя одномерный мир, являющийся прямой, на котором располагаются N городов, последовательно пронумерованных от 0 до $N-1$. Направление в сторону от первого города к нулевому названо западным, а в обратную — восточным.

Когда в Лайнландии неожиданно начался кризис, все были жители мира стали испытывать глубокое смятение. По всей Лайнландии стали ходить слухи, что на востоке живётся лучше, чем на западе.

Так и началось Великое Лайнландское переселение. Обитатели мира целыми городами отправились на восток, покинув родные улицы, и двигались до тех пор, пока не приходили в город, в котором средняя цена проживания была меньше, чем в родном.

Формат входных данных

В первой строке дано одно число N ($2 \leqslant N \leqslant 10^5$) — количество городов в Лайнландии.

Во второй строке дано N чисел a_i ($0 \leqslant a_i \leqslant 10^9$) — средняя цена проживания в городах с нулевого по $(N-1)$-й соответственно.

Формат выходных данных

Для каждого города в порядке с нулевого по $(N-1)$-й выведите номер города, в который переселятся его изначальные жители. Если жители города не остановятся в каком-либо другом городе, отправившись в Восточное Бесконечное Ничто, выведите -1.

Пример

стандартный ввод	стандартный вывод
10 1 2 3 2 1 4 2 5 3 1	-1 4 3 4 -1 6 9 8 9 -1

Задача B. Минимум на отрезке

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Рассмотрим последовательность целых чисел длины N. По ней с шагом 1 двигается "окно"длины K, то есть сначала в "окне"видно первые K чисел, на следующем шаге в "окне"уже будут находиться K чисел, начиная со второго, и так далее до конца последовательности. Требуется для каждого положения "окна"определить минимум в нём.

Формат входных данных

В первой строке входных данных содержатся два числа N и K $(1 \leqslant N \leqslant 150000, 1 \leqslant K \leqslant 10000, K \leqslant N)$ — длины последовательности и "окна соответственно. На следующей строке находятся N чисел — сама последовательность.

Формат выходных данных

Выходые данные должны содержать $N - K + 1$ строк — минимумы для каждого положения "окна".

Пример

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

Задача C. ORные подотрезки

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

Однажды вы нашли глубоко в шкафу массив, состоящий из n целых чисел $a_1, a_2, \ldots, a_n$.

Назовем *показателем ORности* подотрезка массива $[l, r]$ $(1 \leqslant l \leqslant r \leqslant n)$ величину $a_l \mid a_{l+1} \mid \ldots \mid a_r$. Здесь $\mid$ означает операцию побитового «ИЛИ». В большинстве современных языков программирования данная операция обозначается как «|».

Вы захотели узнать количество подотрезков массива, *показатель ORности* которых не меньше, чем ваше любимое число k.

Формат входных данных

Первая строка содержит два целых числа n и k $(1 \leqslant n \leqslant 2 \cdot 10^5,\ 0 \leqslant k \leqslant 10^9)$ — количество элементов в массиве, а также минимальный *показатель ORности* рассматриваемых подотрезков.

Вторая строка содержит n целых чисел $a_1, a_2, \ldots, a_n$ $(0 \leqslant a_i \leqslant 10^9)$ — элементы массива.

Формат выходных данных

Выведите одно целое число — количество подотрезков, *показатель ORности* которых не меньше, чем k.

Система оценки

Данная задача состоит из трех подзадач. Баллы за каждую подзадачу начисляются только в том случае, если все тесты данной и предыдущих подзадач успешно пройдены.

Подзадача 0 (0 баллов). Тесты из условия.

Подзадача 1 (20 баллов). $1 \leqslant n \leqslant 100$.

Подзадача 2 (30 баллов). $1 \leqslant n \leqslant 2\,500$.

Подзадача 3 (50 баллов). Нет дополнительных ограничений.

Примеры

стандартный ввод	стандартный вывод
5 12 1 16 2 8 4	10
3 0 5 5 5	6

Замечание

В первом примере есть 10 подходящих подотрезков:

1. $[1, 2]$: $a_1 \mid a_2 = 17$
2. $[1, 3]$: $a_1 \mid a_2 \mid a_3 = 19$
3. $[1, 4]$: $a_1 \mid a_2 \mid a_3 \mid a_4 = 27$
4. $[1, 5]$: $a_1 \mid a_2 \mid a_3 \mid a_4 \mid a_5 = 31$
5. $[2, 2]$: $a_2 = 16$
6. $[2, 3]$: $a_2 \mid a_3 = 18$
7. $[2, 4]$: $a_2 \mid a_3 \mid a_4 = 26$
8. $[2, 5]$: $a_2 \mid a_3 \mid a_4 \mid a_5 = 30$
9. $[3, 5]$: $a_3 \mid a_4 \mid a_5 = 14$

10. $[4, 5]$: $a_4 \mid a_5 = 12$

Во втором примере подходят все подотрезки массива.

Задача D. Колонизаторы - 2

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Недавно вышла новая версия *легендарной* настольной игры «Колонизаторы»! Вы со своими друзьями, конечно же, не упустите шанс поиграть в новую игру.

Одной из основных механик игры является событие «Грабёж». Пусть сейчас играет n человек, пронумерованных слева направо числами от 1 до n. У i-го игрока в руке находятся a_i карт. В момент, когда происходит событие «Грабёж», происходит следующее:

1. Сначала все игроки одновременно выполняют действие: найти ближайшего игрока справа, у которого в руке больше карт, и отдать ему одну карту.

2. Затем все игроки одновременно выполняют действие: найти ближайшего игрока слева, у которого в руке больше карт, и отдать ему одну карту.

Если у некоторого игрока нет необходимого соседа слева или справа, либо у него в руке нет ни одной карты, действие для данного игрока пропускается.

Вы с друзьями уже начали игру, и вот, внезапно, одно неаккуратное действие привело к событию «Грабёж». Выясните, какое максимальное количество карт в руке окажется у какого-то игрока после этого грабежа.

Формат входных данных

В первой строке записано целое число n $(1 \leqslant n \leqslant 100\,000)$ — количество игроков.

Во второй строке через пробел записаны n целых чисел $a_1, a_2, \ldots, a_n$ $(0 \leqslant a_i \leqslant 10^9)$ — количество карт в руках игроков.

Формат выходных данных

Выведите одно целое число — максимальное количество карт в руке после события «Грабёж».

Система оценки

Номер подзадачи	Баллы	Дополнительные ограничения	Необходимые подзадачи
0	0	Тесты из условия	
1	42	$1 \leqslant n \leqslant 100$	
2	58	Нет дополнительных ограничений	1

Примеры

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

Замечание

Рассмотрим первый пример.

Количество карт у игроков после первой фазы «Грабежа»: $0, 4, 1, 1, 5$.

Количество карт у игроков после второй фазы «Грабежа»: $0, 6, 0, 0, 5$.

Максимальное количество карт после «Грабежа» — 6.

Во втором примере $n = 1$, поэтому единственный игрок никому не отдаст свои карты.

Тесты к этой задаче состоят из 2 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов необходимых групп. Тесты из условия не оцениваются.

Задача E. Наибольший общий делитель

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Наибольшим общим делителем непустого набора натуральных чисел A называется максимальное натуральное число d, такое что оно является одновременно делителем всех чисел множества A.

Задан массив натуральных чисел $[a_1, a_2, \ldots, a_n]$ и число k. Требуется выбрать в нем подмассив из k подряд идущих элементов $[a_l, a_{l+1}, \ldots, a_{l+k-1}]$, чтобы их наибольший общий делитель был как можно больше, и вывести этот наибольший общий делитель.

Формат входных данных

Первая строка ввода содержит два целых числа n и k ($2 \leqslant n \leqslant 500\,000$, $2 \leqslant k \leqslant n$).

Вторая строка содержит n натуральных чисел $a_1, a_2, \ldots, a_n$ ($1 \leqslant a_i \leqslant 10^{18}$).

Формат выходных данных

Выведите одно натуральное число — максимальное возможное значение наибольшего общего делителя элементов подмассива длины k заданного массива.

Примеры

стандартный ввод	стандартный вывод
10 4 2 3 4 8 12 6 12 18 4 3	6
2 2 12 18	6
3 2 12 18 24	6

Задача F. Большой, белый, очень прямоугольный

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	256 мегабайт

В прямоугольной таблице клетки раскрашены в белый и черный цвета. Найти в ней прямоугольную область белого цвета, состоящую из наибольшего количества ячеек.

Формат входных данных

Во входных данных записана сначала высота N, а затем ширина M таблицы ($1 \leqslant N \leqslant 5000$, $1 \leqslant M \leqslant 5000$), а затем записано N строк по M чисел в каждой строке, где 0 означает, что соответствующая клетка таблицы выкрашена в белый цвет, а 1 — что в черный.

Формат выходных данных

В выходной файл вывести одно число — количество клеток, содержащихся в наибольшем по площади белом прямоугольнике.

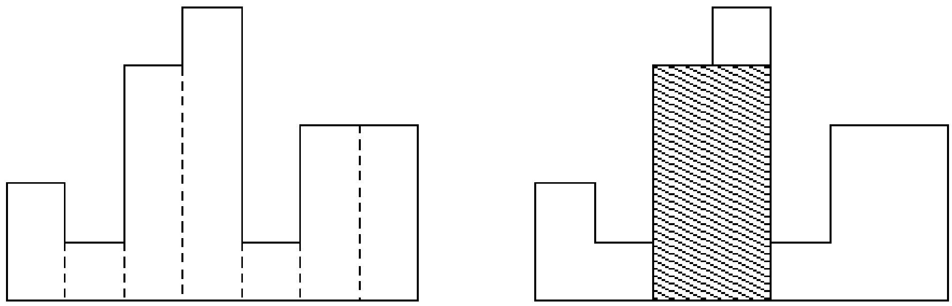
Примеры

стандартный ввод	стандартный вывод
5 6 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0	9
4 4 0 0 0 0 0 1 0 1 0 0 0 0 1 1 0 0	4

Задача G. Гистограмма

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Гистограмма является многоугольником, сформированным из последовательности прямоугольников, выровненных на общей базовой линии. Прямоугольники имеют равную ширину, но могут иметь различные высоты. Например, фигура слева показывает гистограмму, которая состоит из прямоугольников с высотами 2, 1, 4, 5, 1, 3, 3. Все прямоугольники на этом рисунке имеют ширину, равную 1.



Обычно гистограммы используются для представления дискретных распределений, например, частоты символов в текстах. Отметьте, что порядок прямоугольников очень важен. Вычислите область самого большого прямоугольника в гистограмме, который также находится на общей базовой линии. На рисунке справа заштрихованная фигура является самым большим выровненным прямоугольником на изображенной гистограмме.

Формат входных данных

В первой строке входного файла записано число N $(0 < N \leqslant 10^6)$ — количество прямоугольников гистограммы. Затем следует N целых чисел $h_1 \ldots h_n$, где $0 \leqslant h_i \leqslant 10^9$. Эти числа обозначают высоты прямоугольников гистограммы слева направо. Ширина каждого прямоугольника равна 1

Пример

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

Задача H. Очередь в магазине

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В одном известном магазине случилась распродажа, однако администрация не учла одну проблему: в магазине всего одна касса! Сразу после начала распродажи возле кассы организовалась длинная очередь. Никто не любит очереди, поэтому у покупателей постепенно возрастает уровень агрессии. От вас требуется рассмотреть процесс продвижения очереди.

Могут происходить события трёх типов:

1. В конец очереди встал человек с уровнем агрессии a;
2. Первый человек в очереди начал ругаться с кассиром, в результате чего уровень его агрессии увеличился на x, а уровень агрессии каждого из **остальных** людей в очереди (если в очереди стоит не один человек) увеличился на y;
3. Первый человек в очереди оплатил покупку и ушёл из магазина.

От вас требуется обработать N событий. Будем считать, что изначально очередь пуста. Так как администрация магазина заботится о своей репутации, им важно знать, насколько агрессивными их покупатели уходят из магазина. Поэтому для каждого события третьего типа нужно определить уровень агрессии человека, который ушёл из магазина.

Формат входных данных

В первой строке записано одно число N — количество событий ($2 \leqslant N \leqslant 300000$).

В каждой из следующих N строк содержится описание очередного события:

- `1 a`, если произошло событие первого типа;
- `2 x y`, если произошло событие второго типа;
- `3`, если произошло событие третьего типа.

Для всех событий верно, что $1 \leqslant a, x, y \leqslant 10^9$. Гарантируется, что события второго и третьего типов происходят только в том случае, если в очереди есть хотя бы один человек. Также гарантируется, что после N событий в очереди не останется ни одного человека. Возможны случаи, когда первый человек в очереди несколько раз подряд ссорится с кассиром.

Формат выходных данных

Для каждого запроса третьего типа выведите одно число — уровень агрессии человека, который ушёл из магазина. Каждое число следует выводить на отдельной строке.

Система оценки

Подзадача	Баллы	Дополнительные ограничения	Оценка	Необходимые подзадачи
0	0	Тесты из условия	подзадача	–
1	40	$2 \leqslant N \leqslant 1000$ Для всех событий $1 \leqslant a, x, y \leqslant 1000$	подзадача	–
2	60	Дополнительных ограничений нет	подзадача	1

Пример

стандартный ввод	стандартный вывод
8 1 4 1 2 2 6 1 3 2 10 20 1 1 3 3	10 13 1

Замечание

Сначала в очередь встали два человека с уровнями агрессии 4 и 2 соответственно. Затем первый человек поссорился с кассиром, после чего уровни агрессии людей стали равны 10 и 3. После этого первый человек ушёл из очереди, а второй поссорился с кассиром. Теперь уровень его агрессии равен 13. Затем в очередь встал человек с уровнем агрессии 1, после чего оба человека ушли из магазина.

Задача I. Ксюша и покемоны

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

Недавно Ксюша снова зашла в свою любимую игру «Покемон идти». Она еще даже не догадывается, что в какой-то момент попытается поймать покемонов там, где не стоит этого делать, но это уже другая история...

Сейчас же она находится на улице и идет на охоту. На улице есть n покемонов, i-й из них имеет красоту a_i. У Ксюши очень мало времени, поэтому она успеет поймать ровно k покемонов.

Известно, что девочка будет ловить этих существ следующим образом. Сначала она поймает некоторого покемона с номером i, после чего поймает покемона с номером $i + 1$, и так далее до тех пор, пока не поймает покемона с номером $i + k - 1$.

Сегодня Ксюша посмотрела в свой шкаф с функциями радости и обнаружила следующую необычную функцию, которую и будет использовать. Если девочка поймала покемонов с номерами от i до $i + k - 1$ включительно, то она получит $\left(\min\limits_{j=i}^{i+k-1} a_j\right) \cdot \left(\sum\limits_{j=i}^{i+k-1} a_j\right)$ радости. Иными словами, количество радости равно минимальной красоте пойманного покемона, умноженной на сумму красот всех пойманных покемонов.

Теперь перед Ксюшей стоит задача — определить, какое максимальное количество радости от ловли покемонов она может получить. А ваша задача — помочь ей!

Формат входных данных

В первой строке записаны два целых числа n и k ($1 \leqslant k \leqslant n \leqslant 3 \cdot 10^5$) — количество покемонов на улице, а также количество покемонов, которых должна поймать Ксюша.

Вторая строка содержит n целых чисел $a_1, a_2, \ldots, a_n$ ($1 \leqslant a_i \leqslant 10^6$) — красоты покемонов.

Формат выходных данных

В единственной строке выведите одно целое число — максимальную радость, которую может получить девочка от ловли покемонов.

Система оценки

Помимо тестов из условия, данная задача содержит 50 тестов, каждый из которых оценивается независимо. За каждый успешно пройденный тест вы получите 2 балла.

Пример

стандартный ввод	стандартный вывод
5 3 1 2 3 4 5	36

Замечание

В первом примере можно поймать покемонов с номерами 3, 4 и 5, получив при этом $3 \cdot (3 + 4 + 5) = 36$ единиц радости.

Задача J. Прыгающий робот

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Компания «Flatland Dynamics» разрабатывает прыгающего робота. Для испытания робота используется полигон, на котором организован круговой маршрут из n специальных платформ, пронумерованных от 1 до n. Расстояние между i-й и $i+1$-й платформой равно d_i, аналогично расстояние между n-й и 1-й платформой равно d_n.

Робот оснащен искусственным интеллектом и в процессе испытания учится прыгать все дальше. В любой момент времени робот характеризуется своей *ловкостью* — целым числом a. Робот может перепрыгнуть с платформы i на платформу $i+1$, если $a \geqslant d_i$. Аналогично, прыжок с n-й платформы на 1-ю возможен, если $a \geqslant d_n$. При этом после каждого прыжка ловкость робота увеличивается на 1.

Разработчики робота выбирают одну из платформ в качестве стартовой. Они считают эксперимент удачным, если робот может, совершив n прыжков от текущей платформы к следующей, завершить полный круг и вернуться на ту же платформу. Разработчикам необходимо выяснить, для какого минимального значения начальной ловкости робота им удастся провести эксперимент и с какой платформы роботу следует начать прыжки.

Формат входных данных

На первой строке ввода находится число n ($3 \leqslant n \leqslant 10^7$).

Вторая строка содержит одно целое число f, которое описывает формат, в котором задан массив расстояний между платформами.

Если $f = 1$, то на третьей строке находятся n целых чисел $d_1, d_2, \ldots, d_n$ ($1 \leqslant d_i \leqslant 10^9$).

Если $f = 2$, то на третьей строке находится число m $\left(2 \leqslant m \leqslant \min(n, 10^5)\right)$ и три целых числа x, y и z ($0 \leqslant x, y, z \leqslant 10^9$). На четвертой строке находятся m целых чисел $c_1, c_2, \ldots, c_m$ ($1 \leqslant c_i \leqslant 10^9$). Значения d_i вычисляются по следующим формулам.

Если $1 \leqslant i \leqslant m$, то $d_i = c_i$.

Если $m+1 \leqslant i \leqslant n$, то $d_i = \left((x \cdot d_{i-2} + y \cdot d_{i-1} + z) \bmod 10^9\right) + 1$.

Здесь mod означает остаток от целочисленного деления, в языках C++, Java и Python он обозначается символом «%».

Формат выходных данных

Требуется вывести два целых числа: минимальную допустимую начальную ловкость a и номер стартовой платформы, на которую можно разместить робота, чтобы успешно провести эксперимент.

Если возможных стартовых платформ для минимальной начальной ловкости несколько, можно вывести любую из них.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой и необходимых подзадач успешно пройдены.

Подзадача	**Баллы**	**Дополнительные ограничения**	**Необх. подзадачи**	**Информация о проверке**
1	15	$n \leqslant 300$, $f = 1$, $d_i \leqslant 300$		первая ошибка
2	17	$n \leqslant 5000$, $f = 1$	1	первая ошибка
3	10	$n \leqslant 100\,000$, $f = 1$, гарантируется, что оптимально начать с первой платформы		первая ошибка
4	20	$n \leqslant 100\,000$, $f = 1$	1–3	первая ошибка
5	5	$f = 2$, гарантируется, что оптимально начать с первой платформы	3	первая ошибка
6	33	$f = 2$	1–5	первая ошибка

Примеры

стандартный ввод	стандартный вывод
5 1 3 7 4 2 5	4 3
10 2 5 1 2 3 1 2 3 4 5	653 1

Замечание

Во втором примере массив расстояний между платформами равен $[1, 2, 3, 4, 5, 18, 45, 112, 273, 662]$. Значения от d_6 до d_{10} вычисляются по формулам:

$d_6 = \big((1 \cdot d_4 + 2 \cdot d_5 + 3) \bmod 10^9\big) + 1 = \big((1 \cdot 4 + 2 \cdot 5 + 3) \bmod 10^9\big) + 1 = 18$
$d_7 = \big((1 \cdot d_5 + 2 \cdot d_6 + 3) \bmod 10^9\big) + 1 = \big((1 \cdot 5 + 2 \cdot 18 + 3) \bmod 10^9\big) + 1 = 45$
$d_8 = \big((1 \cdot d_6 + 2 \cdot d_7 + 3) \bmod 10^9\big) + 1 = \big((1 \cdot 18 + 2 \cdot 45 + 3) \bmod 10^9\big) + 1 = 112$
$d_9 = \big((1 \cdot d_7 + 2 \cdot d_8 + 3) \bmod 10^9\big) + 1 = \big((1 \cdot 45 + 2 \cdot 112 + 3) \bmod 10^9\big) + 1 = 273$
$d_{10} = \big((1 \cdot d_8 + 2 \cdot d_9 + 3) \bmod 10^9\big) + 1 = \big((1 \cdot 112 + 2 \cdot 273 + 3) \bmod 10^9\big) + 1 = 662$