Задача A. Ремонт кладовки

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

Дима купил кладовку размера $X \times Y \times Z$, где X, Y, Z — это длина, ширина и высота в метрах соответственно. Но она оказалась без окон, без дверей и с голыми стенами. В магазине продается два типа обоев. В наличии имеется S_1 квадратных метров обоев первого типа, стоимостью C_1 рублей за квадратный метр, а второго типа — S_2 квадратных метров стоимостью C_2 рублей за квадратный метр.

Дима хочет сделать дверь размера $A \times B$, где A — ширина, а B — высота, в одной из стен (обои на дверь клеить не надо). Также он хочет, чтобы на стенах, расположенных друг напротив друга, были наклеены одинаковые обои. То есть обе стены размером $X \times Z$ должны быть оклеены одним типом обоев. Аналогично, обе стены размером $Y \times Z$ также должны быть оклеены одним типом обоев. Определите, получится ли у него поклеить обои, и если получится, то какая минимальная сумма в рублях ему потребуется.

Формат входных данных

В первой строке вводится три целых числа X, Y и Z ($1 \leqslant X, Y, Z \leqslant 10\,000$) — длина, ширина и высота комнаты.

Во второй строке вводится четыре целых числа S_1, C_1, S_2 и C_2 ($1 \leqslant S_1, C_1, S_2, C_2 \leqslant 10^8$) — количество квадратных метров обоев первого типа на складе, стоимость квадратного метра обоев первого типа, количество квадратных метров обоев второго типа и стоимость квадратного метра обоев второго типа.

В третьей строке вводится два числа A и B ($1 \leqslant A, B \leqslant 10\,000$) — ширина и высота двери.

Формат выходных данных

Определите, сможет ли Дима оклеить кладовку обоями. Если это невозможно, то выведите -1. Иначе выведите минимальную сумму в рублях, которую Дима потратит на покупку обоев.

Система оценки

Решения, верно работающие при $X = Y = Z$, будут оцениваться не менее чем в 30 баллов.

Примеры

стандартный ввод	стандартный вывод
6 5 10 300 10 10 1 2 3	2140
6 5 10 200 10 95 1 2 3	1294
5 5 4 100 1 100 1 2 5	-1
10 10 10 150 5 150 5 1 2	-1

Замечание

В первом примере Дима установит дверь в стену размером 5×10 и наклеит первый вид обоев на все стены.

Во втором примере Дима установит дверь в стену 5×10, наклеит первый вид обоев на стены 6×10 и второй вид обоев на стены 5×10.

В третьем примере высота двери слишком большая.

В четвертом примере суммарная площадь доступных обоев меньше, чем площадь стен.

Задача B. Цветочный магазин

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

Петя открыл цветочный магазин. Магазин Пети занимается изготовлением и продажей букетов. Всего существует n видов цветов, занумерованных от 1 до n. Каждый букет, чтобы быть гармоничным и красивым, должен состоять из цветов всех видов, по одной штуке каждого вида. В магазине уже есть a_i штук цветов вида i. На цветочной базе можно купить цветок любого вида за 1 рубль.

Определите, сколько букетов сможет собрать Петя, если потратит не более x рублей на покупку цветов на базе. Ответьте на q запросов с различными x_i.

Формат входных данных

В первой строке входных данных находятся два целых числа n и q ($1 \leqslant n, q \leqslant 10^5$) — количество различных типов цветов и количество запросов.

Во второй строке находятся n целых чисел $a_1, a_2, \cdots, a_n$ ($0 \leqslant a_i \leqslant 10^9$) — количество цветов каждого вида, имеющихся в магазине.

В третьей строке находятся q целых чисел $x_1, x_2, \cdots, x_q$ ($0 \leqslant x_i \leqslant 10^9$) — запросы Пети.

Формат выходных данных

Выходной файл должен содержать q чисел, где i-е число это максимальное количество букетов, которое можно собрать потратив не более x_i рублей.

Система оценки

Задача состоит из 20 тестов, не считая тестов из условия. Все тесты можно разделить на следующие группы:

Группа	Макс. балл	Доп. ограничения	Комментарий
0	0	–	Тесты из условия
1	18	Все a_i равны между собой	
2	30	$n, q \leqslant 100$, $a_i, x_i \leqslant 100$	
3	28	$n \leqslant 1000$, $q \leqslant 100$	
4	24	–	

Примеры

стандартный ввод	стандартный вывод

`2 3`
`1 0`
`1 2 5`
	`1 1 3`

`5 6`
`1 2 1 3 7`
`3 1 5 10 15 20`
	`2 1 3 4 5 6`

Замечание

В первом примере у Пети изначально есть 1 цветок первого типа и 0 цветов второго типа.

В первом запросе у него есть 1 рубль, он покупает цветок второго типа и делает 1 букет.

Во втором запросе у него есть 2 рубля, он не может сделать два букета за 2 рубля, поэтому ответ по прежнему 1.

В третьем запросе у него есть 5 рублей, он покупает 2 цветка первого типа и 3 цветка второго типа и делает 3 букета.

Задача C. Операции с числом

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

Дима работает на складе чисел. Он входит на склад с двоичным числом $x = 0$. Ему необходимо превратить свое число x в число s. Для этого на складе есть два автомата для увеличения чисел.

Первый автомат увеличивает двоичное число x на 1 за a секунд. Он расположен слева от входа на склад, в p секундах ходьбы от входа.

Второй автомат умножает двоичное число x на 2 за b секунд. Он расположен справа от входа на склад, в q секундах ходьбы от входа.

Таким образом, если Диме понадобится дойти от одного автомата до другого, он потратит $p + q$ секунд. Исходно он находится у входа на склад.

Помогите Диме узнать, за какое наименьшее количество секунд можно получить число $x = s$ и вернуться ко входу на склад.

Число в двоичной системе счисления из n цифр, представимое в виде: $\overline{a_1 a_2 \ldots a_n}$ ($a_i \in \{0, 1\}$), равно $2^{n-1} \cdot a_1 + 2^{n-2} \cdot a_2 + \ldots + 2 \cdot a_{n-1} + a_n$. ($a_1 = 1$ при $n > 1$, то есть число не имеет ведущих нулей).

Формат входных данных

В первой строке даны два целых числа a и b в десятичной записи ($1 \leqslant a, b \leqslant 10^9$) — время, которое потребуется автоматам для увеличения числа.

Во второй строке даны целые числа p и q в десятичной записи ($0 \leqslant p, q \leqslant 10^9$) — расстояние от входа на склад до первого и второго автоматов.

В третьей строке дано число s в двоичной системе счисления без ведущих нулей (кроме случая $s = 0$). Длина числа s не превышает 100 000 цифр.

Формат выходных данных

Выведите минимальное количество секунд, которое потребуется, чтобы из $x = 0$ получить $x = s$, пользуясь автоматами, и вернуться ко входу на склад.

Система оценки

Задача состоит из 25 тестов, не считая тестов из условия. Каждый тест оценивается независимо в 4 балла. Все тесты можно разделить на следующие группы:

Номер	Макс. балл	Доп. ограничения
1	20	Размер s не больше 5
2	20	Размер s не больше 20
3	20	$p = q = 0$
4	20	Размер s не больше 1000
5	20	Размер s не больше 100 000

Примеры

стандартный ввод	стандартный вывод

`1 2`
`2 3`
`101011`
	`28`

`10 20`
`30 40`
`0`
	`0`

Замечание

В первом тесте необходимо получить число $s = 32 + 8 + 2 + 1 = 43$ в десятичной записи.

Оптимальная последовательность действий: Дима идет к первому автомату (2 секунды), прибавляет к числу единицу 5 раз (5 секунд), потом идет ко второму автомату ($2 + 3 = 5$ секунд), умножает число 3 раза ($3 \cdot 2 = 6$ секунд) и получает число 40, возвращается к первому автомату ($3 + 2 = 5$ секунд), прибавляет единицу 3 раза (3 секунды), и идет ко входу на склад (2 секунды). Всего потрачено 28 секунд.

Во втором тесте у Димы с самого начала есть число $x = 0$.

Задача D. Модульный граф

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

Дан связный неориентированный граф из n вершин. Изначально в i-й вершине ($1 \leqslant i \leqslant n$) записано целое положительное число a_i. Боб хочет за минимальное время попасть из вершины с номером 1 в вершину с номером n. Время, которое требуется, чтобы пройти по ребру, соединяющему вершины u и v, составляет $|a_u - a_v|$.

Перед тем, как начать обход, Боб может поменять значение a_i в не более чем k вершинах. За какое минимальное время можно попасть из 1 в n, поменяв значения оптимальным образом?

Формат входных данных

В первой строке входных данных записаны три целых числа n, m, k ($2 \leqslant n \leqslant 2000, 1 \leqslant m \leqslant 2000, 0 \leqslant k \leqslant 10$) — количество вершин графа, количество ребер и параметр k соответственно.

В следующей строке содержится n целых чисел a_i ($0 \leqslant a_i \leqslant 10^9$) — начальные значения в вершинах.

В следующих m строках содержатся пары чисел u_i, v_i ($1 \leqslant u_i, v_i \leqslant n$) — вершины, соединенные i-м ребром. Гарантируется, что граф связный, в нем нет петель и кратных ребер.

Формат выходных данных

В единственной строке выходных данных выведите ответ на задачу.

Система оценки

Решения, верно работающие в случае, когда все $a_i \leqslant 20$, будут получать не менее 35 баллов.

Решения, верно работающие в случае, когда $m = n - 1$ и существует ребро между вершинами i и $i + 1$ ($1 \leqslant i \leqslant n - 1$), будут получать не менее 29 баллов.

Решения, верно работающие в случае, когда граф не содержит циклов, будут получать не менее 47 баллов.

Решения, верно работающие в случае, когда $k = 0$, будут получать не менее 29 баллов.

Решения, верно работающие в случае, когда $n \leqslant 300$, будут получать не менее 60 баллов.

Примеры

<table>
<tr><th>стандартный ввод</th><th>стандартный вывод</th></tr>
<tr><td>6 7 1
1 15 5 10 4 10
1 2
2 3
3 6
1 4
4 5
5 6
2 5</td><td>9</td></tr>
<tr><td>6 7 3
1 15 5 10 4 10
1 2
2 3
3 6
1 4
4 5
5 6
2 5</td><td>0</td></tr>
</table>

Замечание

В первом тестовом примере одним из оптимальных способов получения ответа может быть изменение значения в вершине с номером 2 с числа 15 на число 3. В таком случае путь $1 \to 2 \to 5 \to 6$ будет иметь стоимость $|1-3|+|3-4|+|4-10|=9$.

Второй пример отличается от первого лишь значением k. Во втором примере можно поменять значения записанные в вершинах с номерами $2, 3, 6$ на 1. Тогда стоимостью пути станет $|1-1|+|1-1|+|1-1|+|1-1|=0$.

Задача E. Караваны и Провинции

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

Далекая страна содержит n городов, соединенных $n-1$ дорогами, при этом из любого города можно добраться до любого другого по дорогам страны.

Известно, что каждый город относится ровно к одной провинции. Город v относится к провинции t_v. Обратите внимание, что конкретная провинция может являться любым подмножеством городов, и возможно из одного города провинции нельзя добраться до другого этой же провинции, проходя только через города этой провинции. Столицей является город номер 1.

Банда разбойников собирается грабить караваны, которые будут идти через города страны. У каждого города есть коэффициент того, насколько удобно в нем грабить. В городе v он равен c_v.

Вам приходят запросы двух типов:

1. Изменить провинцию, к которой относится город v, на t_{new}

2. В k городах с номерами $a_1, a_2, \ldots, a_k$ появляется по одному каравану, которые идут в столицу (город с номером 1) по кратчайшему пути. Разбойники выбирают один город, который находится в провинции t, после чего грабят все караваны, которые пройдут через этот город. Если разбойники ограбят караваны в городе с номером v, то они получат $c_v \cdot num_v$, где c_v — коэффициент города v, а num_v это количество караванов, проходящих через этот город.

Определите максимальный ущерб, равный количеству награбленного разбойниками, для каждого запроса второго типа. Если в провинции, указанной в запросе, нет ни одного города, то ответ на этот запрос равен 0.

Формат входных данных

В первой строке даны два целых числа n и q ($2 \leqslant n \leqslant 200\,000, 1 \leqslant q \leqslant 200\,000$) — количество городов и количество запросов.

Во второй строке дано $n-1$ целое число $p_2, p_3, \ldots, p_n$ ($1 \leqslant p_i < i$), где число p_i означает, что существует дорога между городами i и p_i.

В третьей строке дано n целых чисел $t_1, t_2, \ldots, t_n$ ($1 \leqslant t_i \leqslant n$) — номера провинций у городов.

В четвертой строке дано n целых чисел $c_1, c_2, \ldots, c_n$ ($1 \leqslant c_i \leqslant 10^9$) — коэффициенты успешности грабежа.

Далее идет q строк описаний запросов. В начале каждой строки дано одно целое число x_i ($1 \leqslant x_i \leqslant 2$) — тип запроса.

1. Если $x_i = 1$, то далее идет два целых числа v и t_{new} ($1 \leqslant v, t_{new} \leqslant n$) — номер города, у которого меняется провинция, и номер его новой провинции.

2. Если $x_i = 2$, то далее идут целые числа t и k, и k целых чисел $a_1, a_2, \ldots, a_k$ ($1 \leqslant t, k, a_i \leqslant n$) — номер провинции, в городе которой можно грабить; количество городов, из которых выходят караваны; и номера городов, из которых входят караваны. Гарантируется, что в одном запросе все a_i различны. Также гарантируется, что сумма k по всем запросам второго типа не превышает $200\,000$.

Формат выходных данных

На каждый запрос второго типа выведите одно число — максимальное число, которое разбойники смогут получить. Если в провинции, указанной в запросе, нет ни одного города, то ответ на этот запрос равен 0.

Система оценки

Все тесты можно разделить на следующие группы:

Группа	Макс. балл	Доп. ограничения	Комментарий
0	0	–	Тесты из условия.
1	18	$n, q, \sum k_i \leqslant 1000$	
2	14	Нет запросов первого типа, а также $t_i = 1$.	
3	10	Нет запросов первого типа.	
4	18	$p_i = i - 1$	
5	18	$p_i = \lfloor \frac{i}{2} \rfloor$	
6	22	–	

Пример

стандартный ввод	стандартный вывод
4 4 1 2 2 1 1 3 3 2 3 10 7 2 3 2 3 4 2 1 2 4 3 1 3 1 2 1 2 3 4	10 6 10

Замечание

В первом запросе караваны идут из городов с номерами 3 и 4 и нужно ограбить их в городе из третьей провинции. Это те же самые города с номерами 3 и 4, через каждый из которых пройдет по одному каравану. Поэтому разбойники ограбят караваны в третьем городе и получат $c_3 \cdot 1 = 10 \cdot 1 = 10$.

Во втором запросе караваны также идут из городов с номерами 3 и 4, но теперь нужно ограбить их в городе из первой провинции. В первой провинции находятся города 1 и 2, через каждый из которых пройдет два каравана. Среди них разбойники выбирают город 2, потому что $c_2 > c_1$ и ответ на этот запрос равен $c_2 \cdot 2 = 3 \cdot 2 = 6$.

В третьем провинция для города 3 изменяется на 1.

В четвертом запросе караваны снова идут из городов с номерами 3 и 4, и нужно ограбить караваны в городе из первой провинции. То есть разбойники могут ограбить караваны в одном из городов с номерами $1, 2$ или 3. Через города с номерами 1 и 2 пройдет два каравана, а через город 3 только один. Разбойникам выгодно ограбить караваны в городе 3 и получить $c_3 \cdot 1 = 10 \cdot 1 = 10$.