

## Задача А. Путешествие

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

Маленький Петя очень любит путешествовать. В стране Берляндия, где он живет, есть  $n$  городов, расположенных на одной прямой. Петя пронумеровал их числами от 1 до  $n$  в порядке увеличения красоты. Петя находится в городе 1 и хочет попасть в город  $n$ . Чтобы не портить впечатления о поездке, он может посещать города только в порядке увеличения номеров (а, следовательно, и красоты).

Для перемещения между городами Петя решил воспользоваться услугами единственной авиакомпании страны — Berland Airlines. Стоимость перелёта из города  $i$  в город  $j$  равна  $c_i \cdot |x_i - x_j| + t_j$ , где  $x_i$  — координата города  $i$ ,  $x_j$  — координата города  $j$ , а  $c_i$  — стоимость единицы самолётного топлива в городе  $i$ , а  $t_j$  — стоимость въезда в город  $j$ .

Чтобы было о чем рассказать друзьям, Петя хочет потратить как можно больше (да-да, именно больше) денег на эту поездку. Помогите ему в этом. Обратите внимание, что Пете не обязательно бывать во всех городах.

### Формат входных данных

Первая строка содержит целое число  $n$  — количество городов в Берляндии ( $1 \leq n \leq 100\,000$ ).

Далее следуют  $n$  строк. Строка с номером  $i$  содержит три целых числа —  $x_i$ ,  $c_i$  и  $t_i$  ( $-10^6 \leq x_i \leq 10^6$ ,  $1 \leq c_i \leq 10^6$ ,  $1 \leq t_i \leq 10^6$ ).

### Формат выходных данных

Выведите искомое наибольшее количество денег, которые Петя может потратить чтобы добраться из города 1 в город  $n$ . Гарантируется, что ответ не превосходит  $10^{12}$ .

### Примеры

стандартный ввод	стандартный вывод
4	123
5 10 2	
0 1 10	
15 3 14	
17 2 3	

## Задача В. Сiel и гондолы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Лиса Сiel зашла в парк аттракционов. И вот, она в очереди на колесо обозрения. В очереди стоит  $n$  людей (хотя нет, скорее лис): мы будем считать, что первая лиса стоит в начале очереди, а  $n$ -я лиса стоит в хвосте очереди.

Всего имеется  $k$  гондол, мы распределяем лис по гондолам следующим образом:

- Когда подплывает первая гондола,  $q_1$  лис переходят из начала очереди в подплывшую гондолу.
- Затем, когда подплывает вторая гондола,  $q_2$  лис из начала оставшейся очереди переходит в эту гондолу.
- ...
- Оставшиеся  $q_k$  лис идут с последней ( $k$ -ю) гондолу.

Обратите внимание, что числа  $q_1, q_2, \dots, q_k$  должны быть положительными. Из условия следует, что  $\sum_{i=1}^k q_i = n$  и  $q_i > 0$ .

Вы знаете как лисам не хочется задерживаться в гондолах с незнакомцами. Итак, Ваша задача — найти оптимальный способ размещения (то есть определить оптимальную последовательность  $q$ ), чтобы угодить всем. Для каждой пары лис  $i$  и  $j$  задано значение  $u_{ij}$ , обозначающее степень незнакомости. Можете считать, что  $u_{ij} = u_{ji}$  для всех  $i, j$  ( $1 \leq i, j \leq n$ ) и что  $u_{ii} = 0$  для всех  $i$  ( $1 \leq i \leq n$ ). Тогда значение незнакомости в гондоле определяется как сумма значений незнакомости между всеми парами лис, которые находятся в этой гондоле. Общее значение незнакомости определяется как сумма значений незнакомости по всем гондолам.

Помогите лисе Сiel найти минимальное возможное значение общей незнакомости при некотором оптимальном распределении лис по гондолам.

### Формат входных данных

В первой строке даны два целых числа  $n$  и  $k$  ( $1 \leq n \leq 4000$  and  $1 \leq k \leq \min(n, 800)$ ) — количество лис в очереди и количество гондол. В следующих  $n$  строках записано по  $n$  целых чисел — матрица  $u$ , ( $0 \leq u_{ij} \leq 9$ ,  $u_{ij} = u_{ji}$  и  $u_{ii} = 0$ ).

Пожалуйста, используйте методы быстрого чтения (например, для Java используйте `BufferedReader` вместо `Scanner`).

### Формат выходных данных

Выведите целое число — минимальное возможное значение общей незнакомости.

## Примеры

стандартный ввод	стандартный вывод
5 2 0 0 1 1 1 0 0 1 1 1 1 1 0 0 0 1 1 0 0 0 1 1 0 0 0	0
8 3 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0	7

## Замечание

В первом примере можно распределить лис вот так: 1, 2 идут в одну гондолу, 3, 4, 5 идут в другую гондолу.

Во втором примере оптимальное распределение таково: 1, 2, 3 | 4, 5, 6 | 7, 8.

## Задача С. Утилитаризм

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 5.5 секунд  
Ограничение по памяти: 256 мегабайт

В британском королевстве есть  $n$  городов, пронумерованных от 1 до  $n$ . Некоторые города связаны двусторонними дорогами. Всего есть ровно  $n-1$  дорога, и между любыми двумя городами есть ровно один путь. Также с каждой дорогой связана некоторая ценность.

Сегодня, чтобы почтить отцов-основателей (их было, кстати,  $k$ ) королевства, нынешний король Лелуш решил выбрать  $k$  различных дорог и подарить по одной дороге каждому из основателей. Чтобы избежать ненужных конфликтов, требуется, чтобы дороги не касались своими концами.

На самом деле, Лелушу не так-то и важно, кто получит какую дорогу. С другой стороны, его волнует суммарная ценность подарочных дорог. Вам нужно помочь Лелушу и выбрать  $k$  дорог, максимизируя эту суммарную ценность.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $k$  ( $2 \leq n \leq 250\,000$ ,  $1 \leq k \leq n-1$ ) — количество городов и количество дорог, которые нужно подарить.

Каждая из следующих  $n-1$  содержит три целых числа  $v_i, u_i, c_i$  ( $1 \leq v_i, u_i \leq n$ ,  $-10^6 \leq c_i \leq 10^6$ ) — концы очередной дороги и её ценность.

### Формат выходных данных

Если раздать дороги нельзя, выведите «Impossible».

Иначе выведите одно число — максимальную суммарную цену  $k$  дорог.

### Примеры

стандартный ввод	стандартный вывод
5 1 1 2 2 2 3 3 2 4 10 4 5 6	10
5 2 1 2 2 2 3 3 2 4 10 4 5 6	9
5 3 1 2 2 2 3 3 2 4 10 4 5 6	Impossible

## Задача D. Петя и прямоугольники

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

Маленький Петя очень любит прямоугольники. Петя дал маме список прямоугольников, которые он хочет получить в подарок на Новый Год. Каждый прямоугольник характеризуется  $w$  и высотой  $h$ . Мама хочет сделать Пете приятное и купить все прямоугольники из его списка. Мама отправилась в магазин и узнала, что цена одного прямоугольника равна его площади. К ее счастью, в магазине действует предновогодняя акция, позволяющая покупать прямоугольники не по одному, а сразу наборами. Стоимость одного набора равна ширине самого широкого прямоугольника, умноженной на высоту самого высокого прямоугольника из этого набора. Обратите внимание, что поворачивать прямоугольники (тем самым меняя местами ширину и высоту) нельзя. Помогите маме Пети купить все прямоугольники из списка ее сына, потратив на это наименьшее количество денег.

### Формат входных данных

В первой строке записано число  $N$  ( $1 \leq N \leq 200\,000$ ) — количество прямоугольников в списке Пети. В каждой из следующих  $N$  строк записаны по 2 целых положительных числа, не превышающих  $10^6$  — ширина и высота очередного прямоугольника.

### Формат выходных данных

Выведите одно число — наименьшее количество денег, которое может потратить мама чтобы купить Пете все прямоугольники из его списка.

### Примеры

стандартный ввод	стандартный вывод
4 100 1 15 15 20 5 1 100	500
5 1 10 2 20 3 30 4 40 10 1	170

## Задача E. Новый год и смена хендла

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Наступает новый год. Это значит, что на codeforces пришла пора менять хендлы. Мишке захотелось поменять свой хендл, но таким образом, чтобы люди не забыли, кто он такой.

Чтобы все получилось так, как он хочет, он решил менять только регистры букв. Более формально, за **одну** смену хендла он может выбрать любой отрезок своего хендла  $[i; i + l - 1]$  и сделать **tolower** или **toupper** ко всем буквам своего хендла на этом отрезке (более формально, поменять все строчные буквы на отрезке на соответствующие прописные или наоборот). Длина  $l$  фиксирована для всех смен.

Так как на codeforces нельзя слишком часто менять хендлы, всего Мишка может сделать не более  $k$  таких операций. Какого **минимального** значения может быть  $\min(lower, upper)$  (где  $lower$  — количество строчных букв, а  $upper$  — количество прописных букв) после оптимальной последовательности смен?

### Формат входных данных

Первая строка входных данных содержит три целых числа  $n, k$  и  $l$  ( $1 \leq n, k, l \leq 10^6, l \leq n$ ) — длину хендла Мишки, количество смен и длину отрезка.

Вторая строка входных данных содержит одну строку  $s$ , состоящую из  $n$  строчных и прописных букв латинского алфавита — хендл Мишки.

### Формат выходных данных

Выведите одно целое число — минимальное значение, которого может достигнуть  $\min(lower, upper)$  после того, как Мишка сменил свой хендл не более  $k$  раз так, как описано в условии задачи.

### Примеры

стандартный ввод	стандартный вывод
7 1 4 PikMike	0
15 2 2 AaAaAaAaAAAAaA	2
14 2 6 aBcdEFGHIJkLMn	0
9 2 2 aAaAAAAaA	1

## Задача F. А ты сортируешь запросы?

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

Вам даётся  $N$  запросов, которые бывают трёх типов:

1. Добавить пару чисел  $(a, b)$  в набор.  $(-10^9 \leq a, b \leq 10^9)$
2. Удалить пару добавленную в запросе  $index$ . (Запросы нумеруются числами от 1 до  $N$  в порядке в котором они идут вводятся в программу.)
3. Вам даётся число  $A$  и требуется найти максимум  $a_i \cdot A + b_i$  по всем парам в наборе.  $(-10^9 \leq A \leq 10^9)$ . Гарантируется, что в наборе есть хоть одна пара в этот момент.

### Формат входных данных

Первая строка содержит целое число  $n$  ( $1 \leq n \leq 3 \cdot 10^5$ ) - количество запросов.

Каждая из следующих  $n$  строк начинается с целого числа  $op$  ( $1 \leq op \leq 3$ ) - типа запроса. Далее, в зависимости от типа запроса, следуют, либо числа  $a_i$  и  $b_i$ , либо число  $index$ , либо число  $A$ .

### Формат выходных данных

Для каждого запроса типа 3 выведите в отдельной строке максимальное значение  $a \cdot A + b$ . Гарантируется, что в наборе есть хоть одна пара в этот момент.

### Примеры

стандартный ввод	стандартный вывод
4	-1000000000000000000
1 -1000000000 0	-999999999999999999
3 1000000000	
1 -1000000000 1	
3 1000000000	

## Задача G. Задача без легенды

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

Дан неубывающий массив  $a$  размера  $n$ , а также стоимостей элементов  $c$ . Вам нужно выбрать подпоследовательность (возможно, пустую) с минимальной стоимостью и вывести эту стоимость. Стоимость подпоследовательности из элементов  $i_1, i_2, \dots, i_k$  это сумма стоимостей ее элементов плюс сумма «расстояний» между ее соседними элементами. «Расстояние» между элементами  $i$  и  $j$ , ( $i < j$ ) — это  $(7a_j - 4a_i)(5a_j - 2a_i)(4a_j - a_i)$ .

### Формат входных данных

В первой строке вводится натуральное число  $n$  ( $1 \leq n \leq 10^6$ ). Во второй строке вводится массив целых чисел  $a$  ( $0 \leq a_i \leq 10^6$ ). В третьей строке вводится массив целых чисел  $c$  ( $-10^{18} \leq c_i \leq 10^{18}$ )

### Формат выходных данных

Выведите одно число — минимальную стоимость подпоследовательности.

### Примеры

стандартный ввод	стандартный вывод
2 0 1 2 -6	-6
3 1 2 4 -6 5 1	-6
5 1 2 2 2 4 0 -2 -1 -5 2	-5
4 1 1 3 5 7 -10 8 -7	-10
5 0 1 4 4 4 -8 2 10 4 5	-8
2 1 1 -100 -100	-173

### Замечание

Так как ответ может быть очень большим, рекомендуется использовать `__int128` или еще что-нибудь.