

Задача А. Зсув вліво

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: `0.25 seconds`
Ліміт використання пам'яті: `256 megabytes`

Задається лінійний масив A розмірністю N . Ваше завдання виконати D операцій зсуву вліво всіх елементів масиву.

При зсуві вліво перший елемент масиву переміщається в кінець масиву, а для всіх інших індекси зменшуються на 1.

Формат вхідних даних

Перший рядок вхідного потоку містить два цілі числа N, D ($1 \leq N \leq 10^5, 1 \leq D \leq N$).

У наступному рядку міститься N цілих чисел A_i ($1 \leq A_i \leq 10^6$), які розділяються пропуском.

Формат вихідних даних

Вивести в одному рядку через пропуск елементи масиву A після D зсувів вліво.

Приклад

standard input	standard output
5 4	5 1 2 3 4
1 2 3 4 5	

Задача В. Відсутні числа

Назва вхідного файлу:	standard input
Назва вихідного файлу:	standard output
Ліміт часу:	0.25 seconds
Ліміт використання пам'яті:	256 megabytes

Митець Нумерос створив композицію, в основі якої два списки чисел, які є перестановками один одного. Він дуже пишався своїм творінням. На жаль, переносячи їх з однієї виставки на іншу, деякі цифри з першого списку загубилися. Допоможіть йому знайти ці числа.

Наприклад, у цьому числовому списку [7,2,5,3,5,3] відсутні деякі числа. Оригінальний список був такий: [7,2,5,4,6,3,5,3]. Відсутні числа [4,6].

Формат вхідних даних

Перший рядок містить ціле число n - кількість чисел у неповному списку.

Другий рядок містить n цілих чисел списку A , які розділяються пропуском.

Третій рядок містить ціле число m - кількість чисел у другому списку.

Четвертий рядок містить m цілих чисел списку B , які розділяються пропуском.

Обмеження

$$1 \leq n, m \leq 2 \cdot 10^5$$

$$n < m$$

$$1 \leq A_i, B_i \leq 10^4$$

$$\max(B_i) - \min(B_i) < 101$$

Формат вихідних даних

Виведіть через пропуск загублені числа у неспадному порядку.

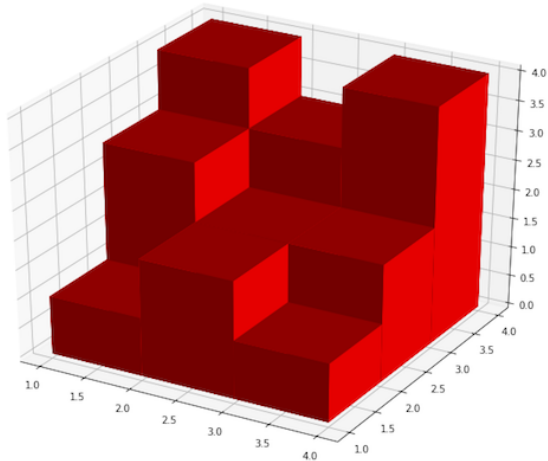
Приклад

standard input	standard output
6	4 6
7 2 5 3 5 3	
8	
7 2 5 6 3 5 4 3	

Задача С. Площа поверхні

Назва вхідного файлу: standard input
Назва вихідного файлу: standard output
Ліміт часу: 0.25 seconds
Ліміт використання пам'яті: 256 megabytes

Орися любить гратися різними іграшками. Її друг Назар працює на фабриці з виробництва іграшок. За основу чергової іграшки Назар взяв дошку A розміром $H \times W$. Дошка розділена на комірки розміром 1×1 , які мають координати (i, j) . В комірці (i, j) записане ціле число A_{ij} . Для створення іграшки Назар ставить A_{ij} кубиків розміром $1 \times 1 \times 1$ на клітинку (i, j) .



Ціна утвореної іграшки буде чисельно дорівнювати площі поверхні іграшки. Складіть програму, яка допоможе Назару визначити ціну іграшки.

Формат вхідних даних

Перший рядок містить два цілі числа H, W ($1 \leq H, W \leq 100$), які розділяються пропуском і позначають розміри дошки.

Наступні H рядків містять W цілих чисел A_{ij} ($1 \leq A_{ij} \leq 100$), розділених пропусками.

Формат вихідних даних

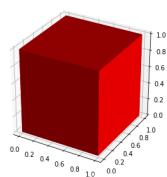
Вивести одне ціле число - ціну іграшки.

Приклади

standard input	standard output
1 1 1	6
3 3 1 3 4 2 2 3 1 2 4	60

Зауваження

Для прикладу 1:



Малюнок з умови відповідає прикладу 2.

Задача D. Найкоротша відстань

Назва вхідного файлу: standard input
Назва вихідного файлу: standard output
Ліміт часу: 0.5 seconds
Ліміт використання пам'яті: 256 megabytes

У деякій країні є N міст. Між містами є лише дороги з одностороннім рухом - вважається, що таким чином кількість автомобільних пригод мінімізується. Знайдіть найкоротшу відстань, яку треба подолати для того, щоб з міста a потрапити в місто b описаними дорогами.

Формат вхідних даних

Перший рядок містить два цілих числа N, M ($2 \leq N \leq 400, 1 \leq M \leq (n \cdot (n - 1))/2$) - кількість міст і кількість доріг відповідно.

Наступні M рядків містять по три числа x, y, d ($1 \leq x, y \leq N, 1 \leq d \leq 350$), де x - місто, з якого йде дорога; y - місто, у яке йде дорога, а d - довжина дороги.

Наступний рядок містить одне ціле число q ($1 \leq q \leq 10^5$) - кількість запитів.

Кожен з наступних q рядків містить по два цілі числа: номер початкового міста і номер кінцевого міста.

З міста до цього ж міста завжди є дорога і довжина її рівна 0.

Якщо від міста a до міста b є декілька прямих доріг з різними довжинами, то вважаємо, що є лише одна дорога із довжиною, яка була введена останньою.

Формат вихідних даних

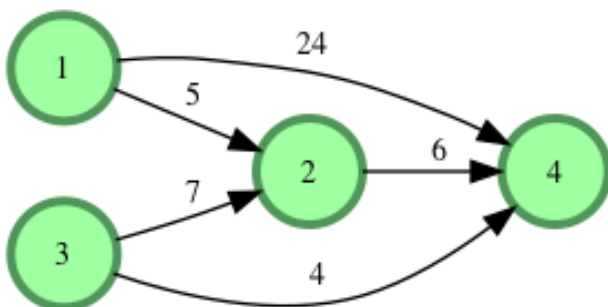
Для кожного запиту вивести довжину найкоротшого шляху з першого міста у друге. У випадку, коли неможливо потрапити з першого міста у друге - виведіть -1.

Приклад

standard input	standard output
4 5	5
1 2 5	-1
1 4 24	11
2 4 6	
3 4 4	
3 2 7	
3	
1 2	
3 1	
1 4	

Зауваження

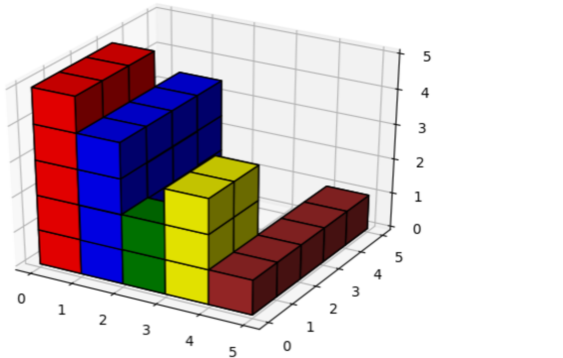
Граф до прикладу з умови:



Задача Е. Тривимірна гістограма

Назва вхідного файлу: standard input
Назва вихідного файлу: standard output
Ліміт часу: 1.5 seconds
Ліміт використання пам'яті: 256 megabytes

Вам надається тривимірна гістограма, яка складається з n блоків, які розміщені поруч один з одним. i -й блок має ширину 1 метр, висоту a_i метри та довжину b_i метри. Іншими словами, спереду це виглядає як гістограма зі стовпчиками висот a_1, a_2, \dots, a_n , а зверху вона виглядає як гістограма з стовпчиками висотами b_1, b_2, \dots, b_n .



Визначте блок з максимальним об'ємом, який можна розмістити всередині заданої тривимірної гістограми. Сторони цього блоку повинні бути паралельними сторонам блоків, з яких складається 3D-гістограма.

Формат вхідних даних

Перший рядок містить ціле число n ($1 \leq n \leq 2 \cdot 10^5$) з опису завдання. Кожен i -й із наступних n рядків містить цілі числа a_i та b_i ($1 \leq a_i, b_i \leq 10^6$) з опису завдання.

Формат вихідних даних

Вивести шуканий об'єм у метрах кубічних

Система оцінки

При $1 \leq n \leq 2000$ - 20 балів
при $1 \leq n \leq 200000$ - 80 балів

Приклад

standard input	standard output
5	24
5 3	
4 4	
2 1	
3 2	
1 5	

Зауваження

На малюнку показана тривимірна гістограма з прикладу вхідних даних. Найбільший блок отримується за допомогою частин перших двох блоків, а він має ширину 2 метри, висоту 4 метри та довжину 3 метри. Об'єм цього блоку становить $2 \times 4 \times 3 = 24$ куб.м.