Общая информация по задачам второго тура

Задача	**Тип задачи**	**Ограничения**
5. Московские числа	стандартная	1 с, 1024 МБ
6. Гаджеты на дереве	стандартная	1 с, 1024 МБ
7. Вырубка деревьев	стандартная	2 с, 1024 МБ
8. Блогеры-путешественники	стандартная	3 с, 1024 МБ

Необходимо считывать данные из стандартного потока ввода. Выходные данные необходимо выводить в стандартный поток вывода.

Баллы за подзадачу, если в условии не указано иное, начисляются только если все тесты этой подзадачи пройдены. Решение запускается на тестах для определенной подзадачи, если все тесты всех необходимых подзадач пройдены.

Для некоторых подзадач может также требоваться, чтобы были пройдены все тесты из условия. Для таких подзадач указана дополнительно буква У.

Задача 5. Московские числа

Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

Скорее всего, вы знакомы с римскими числами. А также наверняка слышали фразу, что Москва — это третий Рим. Поэтому мы решили по аналогии с римскими числами придумать их продвинутую версию — *московские числа.*

Цифрами московского числа являются заглавные английские буквы от A до Z. Числом является строка из нескольких цифр. Каждой цифре сопоставим значение:

A	1	H	$5 \cdot 10^3$	O	10^7	V	$5 \cdot 10^{10}$
B	5	I	10^4	P	$5 \cdot 10^7$	W	10^{11}
C	10	J	$5 \cdot 10^4$	Q	10^8	X	$5 \cdot 10^{11}$
D	50	K	10^5	R	$5 \cdot 10^8$	Y	10^{12}
E	100	L	$5 \cdot 10^5$	S	10^9	Z	$5 \cdot 10^{12}$
F	500	M	10^6	T	$5 \cdot 10^9$		
G	10^3	N	$5 \cdot 10^6$	U	10^{10}		

Значение числа равно сумме вкладов цифр, из которых оно состоит. Вклад цифры бывает как положительным, так и отрицательным. Если правее цифры в числе нет строго большей цифры, то вклад этой цифры равен её значению. Иначе вклад равен её значению, взятому со знаком минус.

Например,

– «BBA» имеет значение $5 + 5 + 1 = 11$;
– «BBBC» имеет значение $-5 + (-5) + (-5) + 10 = -5$;
– «ABC» имеет значение $-1 + (-5) + 10 = 4$;
– «BAC» имеет значение $-5 + (-1) + 10 = 4$;
– «ACA» имеет значение $-1 + 10 + 1 = 10$.

Вам даны несколько заготовок чисел. Каждая заготовка представляет собой строку из заглавных английских букв и знаков вопроса. Для каждой заготовки необходимо определить, какое максимальное число может получиться, если каждый знак вопроса заменить на цифру московского числа.

Формат входных данных

В первой строке дано одно целое число t — количество заготовок ($1 \leqslant t \leqslant 50\,000$).

В следующих t строках даны строки s_i, состоящие из заглавных английских букв и символов «?» — заготовки для чисел. Сумма длин строк s_i не превышает $300\,000$.

Формат выходных данных

Для каждой заготовки выведите две строки. В первой из них выведите в десятичной системе счисления максимальное значение числа, которое может получиться из этой заготовки. А во второй строке — саму заготовку, у которой знаки вопроса заменены на буквы английского алфавита таким образом, чтобы достигалось максимальное значение.

Пример

стандартный ввод	стандартный вывод
4 BBBC ???? A?B?C?D YYYYY?	-5 BBBC 20000000000000 ZZZZ 15000000000034 AZBZCZD 6000000000000 YYYYYY

Система оценивания

Обозначим сумму длин строк s_i как S.

Подзадача	Баллы	Ограничения		Необх. подзадачи	Информация о проверке
		S	дополнительно		
1	6	$S \leqslant 1000$	s_i не содержит «?»		первая ошибка
2	9	$S \leqslant 3 \cdot 10^5$	s_i не содержит «?»	1	первая ошибка
3	40	$S \leqslant 1000$	s_i содержит не более трёх «?»	1	первая ошибка
4	20	$S \leqslant 1000$	–	У, 1, 3	первая ошибка
5	25	$S \leqslant 3 \cdot 10^5$	–	У, 1–4	первая ошибка

Задача 6. Гаджеты на дереве

Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

Вася недавно придумал новое развлечение. Пусть дан связный ориентированный граф, состоящий из n вершин и $2 \cdot (n-1)$ рёбер, причём для каждого ребра (u, v), существует ребро (v, u). Иначе говоря, граф был получен из дерева, в котором каждое ребро было расщеплено на два противоположных ребра, ориентированных в разные стороны.

Назовем *гаджетом* такую пару рёбер (e_1, e_2), что конец e_1 совпадает с началом e_2 или наоборот (в частности, два противоположных друг другу ребра — гаджет). Вася развлекается тем, что разбивает рёбра графа на непересекающиеся гаджеты. Конечно, ему легко удалось это сделать с исходным графом.

Васин друг Петя удалил из дерева $2 \cdot k$ ориентированных рёбер. Таким образом, в графе осталось $m = 2 \cdot (n-1) - 2 \cdot k$ ориентированных ребер.

Теперь Вася хочет узнать, можно ли разбить оставшиеся ребра на непересекающиеся гаджеты, и, если можно, — найти это разбиение. Помогите ему!

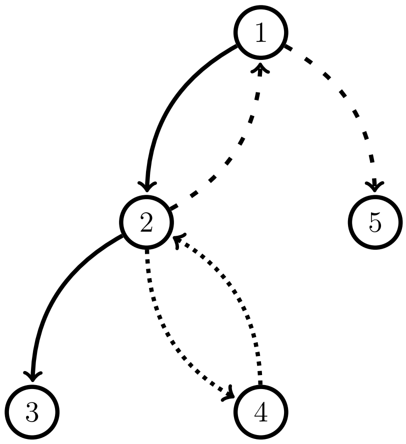


Рис. 1: Разбиение на гаджеты в первом примере

Формат входных данных

В первой строке даны два целых числа n и m ($2 \leqslant n \leqslant 3 \cdot 10^5$, $2 \leqslant m \leqslant 2n-2$) — число вершин и число оставшихся рёбер. Гарантируется, что число m чётное.

В следующих m строках даны по два числа u_i, v_i ($1 \leqslant u_i, v_i \leqslant n$) — начала и концы оставшихся рёбер.

Формат выходных данных

Если разбить рёбра на гаджеты нельзя, выведите «`No`».

В противном случае выведите «`Yes`», а затем выведите $\frac{m}{2}$ строк по 4 числа в каждой — пары рёбер в каждом из гаджетов. Каждое ребро описывается двумя числами: своим началом и концом.

Примеры

стандартный ввод	стандартный вывод

`5 6`
`1 2`
`2 1`
`1 5`
`2 3`
`2 4`
`4 2`
	`Yes`
	`1 2 2 3`
	`2 1 1 5`
	`2 4 4 2`

`4 4`
`2 1`
`2 3`
`2 4`
`4 2`
	`No`

`4 4`
`1 2`
`2 1`
`3 4`
`4 3`
	`Yes`
	`1 2 2 1`
	`3 4 4 3`

Замечание

Разбиение на гаджеты в первом примере изображено на рисунке в условии.

Обратите внимание, что в этой задаче размер входных данных может быть большим. Рекомендуем ознакомиться с разделом «Скорость ввода и выбор ОС» в памятке участника.

Система оценивания

Подзадача	Баллы	Доп. ограничения		Необходимые подзадачи	Информация о проверке
		n	m		
1	7	$n \leqslant 20$	$m \leqslant 20$	У	первая ошибка
2	10	$n \leqslant 200$		У, 1	первая ошибка
3	11	$n \leqslant 3000$	$m = 2n - 4$		первая ошибка
4	29	$n \leqslant 3000$		У, 1–3	первая ошибка
5	11	$n \leqslant 150\,000$	$m = 2n - 4$	3	первая ошибка
6	32	$n \leqslant 150\,000$		У, 1–5	первая ошибка

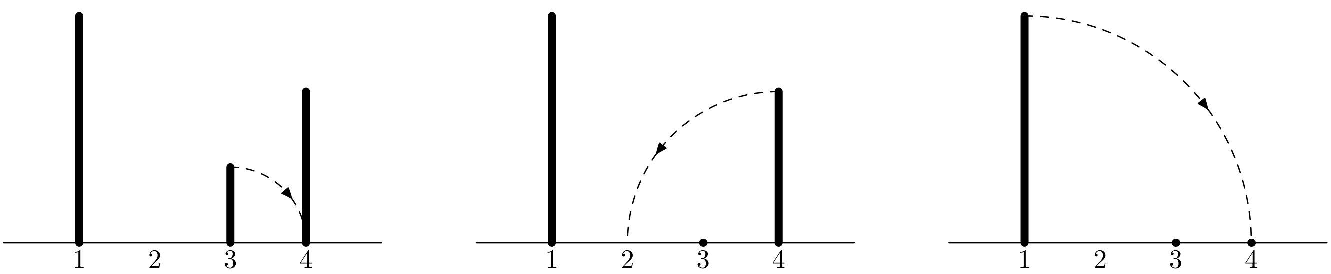
Задача 7. Вырубка деревьев

Ограничение по времени:	2 секунды
Ограничение по памяти:	1024 мегабайта

Организация «2Д» занимается выдачей разрешений на вырубку леса. Ей поступают заявки на вырубку деревьев, расположенных вдоль шоссе. На шоссе расположены n деревьев, i-е из них растёт в точке с координатой x_i и имеет высоту h_i. Информация про имеющиеся деревья упорядочена так, что выполняется $x_1 < x_2 < \ldots < x_{n-1} < x_n$.

Деревья можно вырубать по одному следующим образом. Дерево срубается под корень и затем должно быть повалено либо влево, либо вправо. Чтобы дерево не повредилось при падении, оно не должно задевать ещё не срубленные деревья на расстоянии меньше своей высоты. Иначе говоря, если дерево в точке x_i высоты h_i валят направо, то не должно быть еще не срубленного дерева с координатой x_j такой, что $x_i < x_j < x_i + h_i$. Если это же дерево валят налево, то не должно быть еще не срубленного дерева с координатой x_j такой, что $x_i - h_i < x_j < x_i$.

Слева от дерева с координатой x_1 и справа от дерева с координатой x_n находятся важные постройки, поэтому валить деревья так, чтобы они падали за пределы отрезка $[x_1, x_n]$ запрещается. Иначе говоря, дерево в точке x_i высоты h_i нельзя валить влево, если $x_i - h_i < x_1$, и нельзя валить вправо, если $x_i + h_i > x_n$.



Ситуация из первого примера: сначала второе дерево валят вправо, затем третье влево влево, и наконец, первое валят вправо

В организацию поступили q заявок на вырубку деревьев. Каждая заявка задается двумя числами l_i и r_i и означает, что заявитель хочет вырубать деревья с номерами от l_i до r_i, включительно.

В процессе выполнения заявки разрешается рубить только деревья с номерами от l_i до r_i. Срубленное дерево разрешается валить в том числе и на территорию левее дерева с номером l_i или правее дерева с номером r_i, но не за пределы отрезка $[x_1, x_n]$. Задевать деревья с номерами вне диапазона от l_i до r_i не разрешается.

Для каждой заявки требуется рассчитать, какое максимальное число деревьев с номерами в указанном диапазоне можно вырубить, не повредив ни одно дерево при падении. Каждая заявка должна быть обработана независимо от остальных.

Формат входных данных

Первая строка содержит два целых числа n и q ($1 \leqslant n, q \leqslant 500\,000$).

Каждая из следующих n строк описывает очередное дерево и содержит два целых числа x_i, h_i ($1 \leqslant x_i \leqslant 10^9$; $1 \leqslant h_i \leqslant 10^9$).

Гарантируется, что $x_1 < x_2 < \ldots < x_n$.

Далее следуют q описаний заявок, по одному в строке. Строка i содержит два целых числа l_i, r_i ($1 \leqslant l_i \leqslant r_i \leqslant n$).

Формат выходных данных

Для каждой заявки выведите максимальное количество деревьев, которое можно вырубить, не повредив в процессе ни одного дерева.

Примеры

стандартный ввод	стандартный вывод
3 3 1 3 3 1 4 2 1 1 2 3 1 3	0 2 3
5 3 1 5 3 1 4 2 5 3 6 1 1 5 5 5 1 1	5 1 0
1 1 100 100 1 1	0

Система оценивания

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
		n, q		
1	15	$n, q \leqslant 100$	У	первая ошибка
2	15	$n, q \leqslant 500$	У, 1	первая ошибка
3	15	$n, q \leqslant 5000$	У, 1–2	первая ошибка
4	5	$n, q \leqslant 10\,000$	У, 1–3	первая ошибка
5	10	$n, q \leqslant 100\,000$	У, 1–4	первая ошибка
6	10	$n, q \leqslant 200\,000$	У, 1–5	первая ошибка
7	30	$n, q \leqslant 500\,000$	У, 1–6	потестовая оценка, только баллы

В седьмой подзадаче 30 тестов, каждый из которых независимо оценивается в один балл. Решение будет протестировано на этих тестах только, если оно прошло все тесты первых шести подзадач.

Задача 8. Блогеры-путешественники

Ограничение по времени:	3 секунды
Ограничение по памяти:	1024 мегабайта

Ян и Татьяна решили стать блогерами-путешественниками и публиковать ролики о поездках по городам своей страны.

В стране есть n городов, пронумерованных от 1 до n. Город 1 — столица их страны. Города соединены m двусторонними дорогами, пронумерованными от 1 до m, каждая из которых соединяет два различных города. При этом одну и ту же пару городов могут соединять несколько различных дорог. Из любого города по дорогам можно доехать до любого другого города страны.

Путешественники планируют отправиться из столицы в какой-то другой город, но пока не выбрали в какой. Маршрут путешествия в город k будет состоять из городов $s_1, s_2, \ldots, s_q$ и дорог $r_1, r_2, \ldots, r_{q-1}$, таких что:

- $s_1 = 1$, $s_q = k$;
- дорога r_i соединяет города s_i и s_{i+1};
- ребята не проезжают по одной и той же дороге дважды, поэтому все r_i различны. Допускается проезжать несколько раз через один и тот же город, в том числе через город 1, где путешествие начинается, и город k, в котором путешествие заканчивается.

Для каждой дороги Ян и Татьяна посчитали длительность ролика, который получится при съемке путешествия по этой дороге, длительность ролика для дороги с номером i равна t_i.

В процессе путешествия каждый из ребят выберет одну из дорог маршрута и снимет ролик, посвящённый этой дороге. При этом Ян любит снимать короткие ролики, поэтому выберет на маршруте дорогу с наименьшим значением t_i, а Татьяна предпочитает длинные ролики, поэтому выберет дорогу с наибольшим значением t_i.

Суммарная длина двух роликов будет равна $\min\limits_{1 \leqslant i \leqslant q-1} t_{r_i} + \max\limits_{1 \leqslant i \leqslant q-1} t_{r_i}$.

Ребята планируют выложить ролики на известную платформу, где большей популярностью пользуются короткие ролики, поэтому они хотят минимизировать суммарную длину двух роликов. Чтобы выбрать конечный город и маршрут для путешествия, блогеры хотят для каждого конечного города k подсчитать минимальную по всем возможным маршрутам из города 1 в город k суммарную длину двух роликов.

Формат входных данных

В первой строке даны два целых числа n, m ($2 \leqslant n \leqslant 300\,000$, $1 \leqslant m \leqslant 300\,000$) — количество городов и дорог.

Следующие m строк содержат описания дорог. В i-й из этих строк находятся три целых числа u_i, v_i, t_i ($1 \leqslant u_i, v_i \leqslant n$, $u_i \neq v_i$, $0 \leqslant t_i \leqslant 10^9$) — номера городов, соединённых дорогой, и длительность ролика про эту дорогу.

Гарантируется, что по имеющимся дорогам можно проехать из любого города в любой другой, возможно, через другие города.

Формат выходных данных

Для каждого $2 \leqslant k \leqslant n$ выведите минимальную суммарную длину роликов Яна и Татьяны для путешествия, заканчивающегося в городе k.

Примеры

стандартный ввод	стандартный вывод
3 3 1 2 2 1 3 1 2 3 1	2 2
7 10 1 2 2 1 2 8 2 3 3 3 4 5 3 5 4 4 5 4 6 5 7 6 4 4 1 7 6 6 7 9	4 5 6 6 6 10
4 4 1 2 2 3 2 0 2 4 3 4 3 1	3 2 2

Замечание

В первом примере возможные оптимальные маршруты:

– $1 \overset{t=1}{\to} 3 \overset{t=1}{\to} 2$. Длина роликов в маршруте $1 + 1 = 2$.
– $1 \overset{t=1}{\to} 3$. Длина роликов в маршруте $1 + 1 = 2$.

Во втором примере возможные оптимальные маршруты:

– $1 \overset{t=2}{\to} 2$. Длина роликов в маршруте $2 + 2 = 4$.
– $1 \overset{t=2}{\to} 2 \overset{t=3}{\to} 3$. Длина роликов в маршруте $2 + 3 = 5$.
– $1 \overset{t=2}{\to} 2 \overset{t=3}{\to} 3 \overset{t=4}{\to} 5 \overset{t=4}{\to} 4$. Длина роликов в маршруте $2 + 4 = 6$.
– $1 \overset{t=2}{\to} 2 \overset{t=3}{\to} 3 \overset{t=4}{\to} 5$. Длина роликов в маршруте $2 + 4 = 6$.
– $1 \overset{t=2}{\to} 2 \overset{t=3}{\to} 3 \overset{t=4}{\to} 5 \overset{t=4}{\to} 4 \overset{t=4}{\to} 6$. Длина роликов в маршруте $2 + 4 = 6$.
– $1 \overset{t=2}{\to} 2 \overset{t=8}{\to} 1 \overset{t=6}{\to} 7$. Длина роликов в маршруте $2 + 8 = 10$.

В третьем примере возможные оптимальные маршруты:

– $1 \overset{t=2}{\to} 2 \overset{t=0}{\to} 3 \overset{t=1}{\to} 4 \overset{t=3}{\to} 2$. Длина роликов в маршруте $0 + 3 = 3$.
– $1 \overset{t=2}{\to} 2 \overset{t=0}{\to} 3$. Длина роликов в маршруте $0 + 2 = 2$.
– $1 \overset{t=2}{\to} 2 \overset{t=0}{\to} 3 \overset{t=1}{\to} 4$. Длина роликов в маршруте $0 + 2 = 2$.

Система оценивания

Подз.	Баллы	Ограничения			Необх. подзадачи	Информация о проверке
		n	m	дополнительно		
1	9	$n \leqslant 300\,000$	$m \leqslant 300\,000$	$m = n - 1$		первая ошибка
2	17	$n \leqslant 300\,000$	$m \leqslant 300\,000$	$t_i = 0$ для всех дорог i из города 1		первая ошибка
3	12	$n \leqslant 300\,000$	$m \leqslant 300\,000$	$t_i = 10^9$ для всех дорог i из города 1		первая ошибка
4	9	$n \leqslant 10$	$m \leqslant 10$	каждая пара городов соединена не более чем одной дорогой		первая ошибка
5	6	$n \leqslant 20$	$m \leqslant 20$	каждая пара городов соединена не более чем одной дорогой	4	первая ошибка
6	6	$n \leqslant 2000$	$m \leqslant 2000$	$\lvert u_i - v_i \rvert = 1$ для всех дорог		первая ошибка
7	9	$n \leqslant 2000$	$m \leqslant 2000$		У, 4–6	первая ошибка
8	8	$n \leqslant 5000$	$m \leqslant 300\,000$		У, 4–7	только баллы
9	10	$n \leqslant 300\,000$	$m \leqslant 300\,000$	для всех a существует дорога между парой городов a и $a+1$; для любой пары дорог i и j, для которых $\lvert u_i - v_i \rvert = 1$ и $\lvert u_j - v_j \rvert > 1$ выполнено $t_i \leqslant t_j$	6	только баллы
10	14	$n \leqslant 300\,000$	$m \leqslant 300\,000$		У, 1–9	только баллы