

Task Collecting Diamonds

 3 sec.  4 MB

В Родопских горах было обнаружено месторождение алмазов. Для упрощения будем считать, что месторождение имеет N залов, пронумерованных целыми числами от 0 до $N - 1$. Между некоторыми залами имеется M односторонних коридоров, так что из каждого зала выходит (начинается) хотя бы один коридор. В каждом коридоре находится определенное количество алмазов, которые можно добыть при прохождении через него. Это количество **не изменяется** при многократном прохождении через коридор — оно остается неизменным.

Возможно, что коридор соединяет зал с самим собой, и между одной и той же парой залов может быть несколько коридоров (возможно, в одном направлении). Также не гарантируется, что залы соединены между собой; т. е. может быть пара залов (x, y) , такая что из x нельзя добраться до y .

Петар пройдет через K коридоров, чтобы добыть алмазы. Он выберет какой-нибудь зал s для начала, затем перейдет в зал, пройдя через коридор, начинающийся из s , и так далее, пока не пройдет ровно K коридоров. Обратите внимание, что он может повторять залы и коридоры, и что количество алмазов, которые он собирает из коридора, не меняется при повторении. Обратите внимание, что всегда будет способ пройти через K коридоров подряд.

Петар выберет s и путь, по которому он будет двигаться, следующим образом: во-первых, он хочет максимизировать количество алмазов, которые он соберет в первом коридоре, через который он пройдет. Из всех таких вариантов он будет максимизировать количество алмазов, которые он соберет из второго коридора. Это повторяется K раз. Т. е. Петар хочет выбрать лексикографически наибольший путь. Он задается вопросом, каково будет общее количество алмазов, которые он соберет, если выберет такой путь. Помогите ему это вычислить.



Implementation details

Вы должны реализовать функцию `calculate_diamonds`:

```
long long int calculate_diamonds(int N, int M, int K,  
    std::vector<int> u, std::vector<int> v, std::vector<int> d)
```

- N : количество залов в месторождении алмазов;
- M : количество коридоров между залами;
- K : количество коридоров, по которым должен пройти Петар;
- u, v, d : векторы из M целых чисел, представляющие начальные залы, конечные залы и количество алмазов для коридоров.



Эта функция будет вызываться один раз для каждого теста и должна возвращать одно число - общее количество алмазов, которые Петр соберет, используя свою стратегию.



Constraints

- $1 \leq N \leq 2\,000$
- $1 \leq M \leq 4\,000$
- $1 \leq K \leq 10^9$
- $0 \leq u[i], v[i] < N$
- $1 \leq d[i] \leq 10^9$ для каждого $0 \leq i < M$
- Гарантируется, что из каждого зала начинается хотя бы один коридор.
- **Обратите внимание на необычное ограничение по памяти в 4 МВ.**



Subtasks

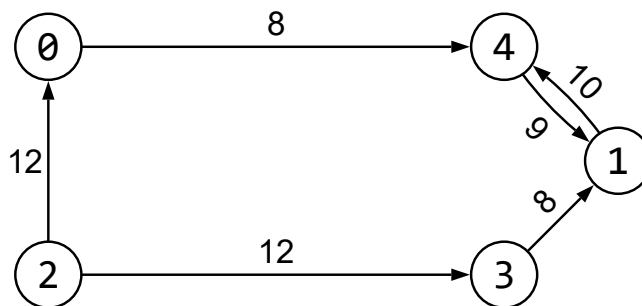
Подзадача	Баллы	Требуемые подзадачи	N	M	K	Дополнительные ограничения
0	0	—	—	—	—	Примеры.
1	11	0	≤ 10	≤ 20	≤ 10	—
2	10	0 — 1	≤ 100	$\leq 1\,000$	≤ 1000	—
3	26	0 — 2	≤ 100	$\leq 1\,000$	$\leq 10^9$	—
4	11	—	$\leq 2\,000$	$= N$	$\leq 10^9$	Каждый зал имеет ровно один коридор, начинающийся от него, и ровно один коридор, заканчивающийся в нем.
5	10	—	$\leq 2\,000$	$\leq 4\,000$	$\leq 10^9$	Все $d[i]$ различны.
6	11	—	$\leq 2\,000$	$\leq 4\,000$	$\leq 10^9$	Существует ровно одно $d[i] = 2$ ($0 \leq i < M$), а все остальные значения в d равны 1.
7	21	0 — 6	$\leq 2\,000$	$\leq 4\,000$	$\leq 10^9$	—



Example 1

Рассмотрим следующий вызов функции и иллюстрацию для $N = 5$, $M = 6$ и $K = 4$:

```
calculate_diamonds(5, 6, 4,
    {2, 0, 4, 2, 3, 1}, {0, 4, 1, 3, 1, 4}, {12, 8, 9, 12, 8, 10})
```



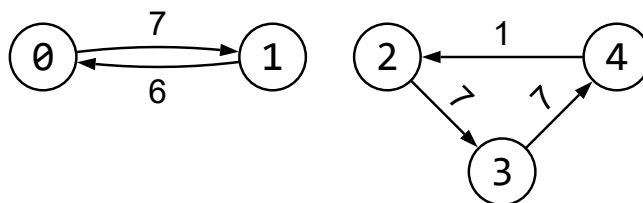
Петар решит пройти по следующим коридорам: $2 \xrightarrow{12} 3 \xrightarrow{8} 1 \xrightarrow{10} 4 \xrightarrow{9} 1$. Общее количество алмазов, которые он соберет, равно 39, и это значение должно быть возвращено вызовом.



Example 2

Рассмотрим следующий вызов функции и иллюстрацию для $N = 5$, $M = 5$ и $K = 4$:

```
calculate_diamonds(5, 5, 4,
    {0, 1, 2, 3, 4}, {1, 0, 3, 4, 2}, {7, 6, 7, 7, 1})
```



Есть 5 вариантов прохождения через 4 коридора:

- (1) $0 \xrightarrow{7} 1 \xrightarrow{6} 0 \xrightarrow{7} 1 \xrightarrow{6} 0$;
- (2) $1 \xrightarrow{6} 0 \xrightarrow{7} 1 \xrightarrow{6} 0 \xrightarrow{7} 1$;
- (3) $2 \xrightarrow{7} 3 \xrightarrow{1} 4 \xrightarrow{7} 2 \xrightarrow{7} 3$;
- (4) $3 \xrightarrow{7} 4 \xrightarrow{1} 2 \xrightarrow{7} 3 \xrightarrow{7} 4$;
- (5) $4 \xrightarrow{1} 2 \xrightarrow{7} 3 \xrightarrow{7} 4 \xrightarrow{1} 2$.

Варианты (2) и (5) не позволяют максимизировать количество алмазов из первого коридора. Из вариантов (1), (3) и (4) только вариант (3) позволяет максимизировать количество алмазов из второго коридора, поэтому это лучший вариант для Петара. Обратите внимание, что вариант (3) не максимизирует количество алмазов из третьего коридора и не максимизирует общее количество алмазов, но это единственная лексикографически наибольшая последовательность. Общее количество алмазов, которые соберет Петр, равно 22, и это должно быть значение, возвращаемое вызовом.



Sample grader

Формат ввода следующий:

- строка 1: три целых числа – значения N , M и K .
- строка $1 + i$: три целых числа $u[i]$, $v[i]$, $d[i]$ – представляющие коридор, начинающийся в зале $u[i]$ и заканчивающийся в зале $v[i]$ с $d[i]$ алмазами для добычи.

Формат вывода следующий:

- строка 1: одно целое число – возвращаемое значение вызова.