

## Задача Збираємо Діаманти

 3 sec.  4 MB

В горах родопи було знайдено родовище діамантів. Припустимо що родовище має  $N$  холів, пронумерованих від 0 до  $N - 1$ . Існує  $M$  односторонніх коридорів які поєднують деякі пари холів. Гарантовано що з кожного з холів виходить хоча б один коридор. Кожен коридор має певну кількість діамантів які можна видобути проходячи через нього. Ця кількість **не змінюється** якщо через нього пройти - вона залишається такою самою кожного наступного разу.

Може бути, що коридор поєднує хол з самим собою, також може бути кілька коридорів між однією парою холів (можливо направлених в ту ж саму сторону). Також, не гарантується, що холи з'єднані; тобто, може бути пара холів  $(x, y)$  така, що в  $y$  не можна потрапити починаючи з  $x$ .

Петро планує пройти через  $K$  коридорів з ціллю видобудку діамантів. Він вибере довільний хол  $s$  як початок шляху. Потім він вибере коридор, який починається там і перейде в хол на іншому кінці. Він продовжить це робити поки не пройде рівно  $K$  коридорів. Зверніть увагу, що на шляху і холи і коридори можуть повторюватися, а кількість діамантів, які він здобуває під час проходження коридору не змінюється при повторенні. Зверніть увагу на те, що завжди існує спосіб пройти  $K$  коридорів послідовно.

Петро вибирає  $s$  і шлях таким чином: Спочатку, він хоче максимізувати кількість діамантів які він видобуде в першому коридорі через який він пройде. Серед всіх таких варіантів, він хоче максимізувати кількість діамантів здобутих в другому коридорі. Так відбувається  $K$  раз. Тобто Петро хоче вибрати лексикографічно найбільший шлях. Його цікавить скільки діамантів він отримає, якщо вибере шлях таким способом. Допоможіть йому визначити це значення.



### Деталі реалізації

Вам потрібно реалізувати `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)
```

- $N$ : кількість холів в родовищі;
- $M$ : кількість коридорів між холами;
- $K$ : кількість коридорів які він пройде;
- $u, v, d$ : вектори з  $M$  чисел, які позначають початкові, кінцеві холи, та кількість діамантів в коридорі.

Ця функція викликається один раз на тест і повертає 1 число - сумарна кількість

діамантів зібрана Петром використовуючи таку стратегію.



## 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 всіх  $0 \leq i < M$
- Гарантовано, що в кожному холі починається хоча б один коридор.
- **Зверніть увагу на незвично малий ліміт пам'яті 4 МВ.**



## Підзадачі

Підзадача	Бали	Залежності	$N$	$M$	$K$	Додаткові обмеження
0	0	—	—	—	—	Приклади.
1	11	0	$\leq 10$	$\leq 20$	$\leq 10$	—
2	10	0 — 1	$\leq 100$	$\leq 1\,000$	$\leq 1000$	—
3	26	0 — 2	$\leq 100$	$\leq 1\,000$	$\leq 10^9$	—
4	11	—	$\leq 2\,000$	$= N$	$\leq 10^9$	В кожному холі починається рівно один коридор. Також в кожному холі закінчується рівно один коридор.
5	10	—	$\leq 2\,000$	$\leq 4\,000$	$\leq 10^9$	Всі $d[i]$ різні.
6	11	—	$\leq 2\,000$	$\leq 4\,000$	$\leq 10^9$	Існує рівно одне $d[i] = 2$ ( $0 \leq i < M$ ) і всі інші $d$ рівні 1.
7	21	0 — 6	$\leq 2\,000$	$\leq 4\,000$	$\leq 10^9$	—



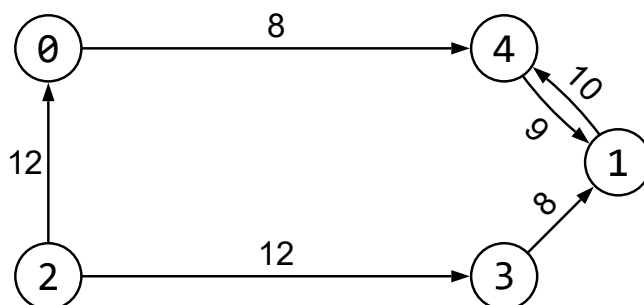
## Приклад 1

Розглянемо наступний приклад, для  $N = 5$ ,  $M = 6$ , та  $K = 4$ :

```
calculate_diamonds(5, 6, 4,
```



$\{2, 0, 4, 2, 3, 1\}, \{0, 4, 1, 3, 1, 4\}, \{12, 8, 9, 12, 8, 10\}$



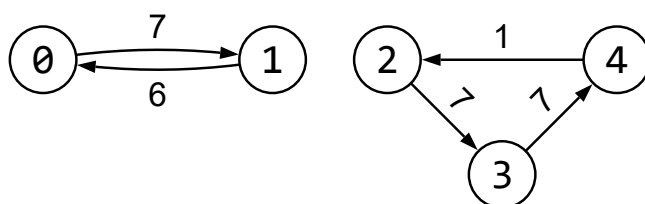
Петро пройде через такі коридори:  $2 \xrightarrow{12} 3 \xrightarrow{8} 1 \xrightarrow{10} 4 \xrightarrow{9} 1$ . Сумарна кількість діамантів яку він отримає 39, це число потрібно повернути з функції.



## Приклад 2

Розглянемо такий приклад, доя  $N = 5$ ,  $M = 5$ , та  $K = 4$ :

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



Є 5 способів пройти через 4 коридори:

- (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)  $4 \xrightarrow{1} 2 \xrightarrow{7} 3 \xrightarrow{7} 4 \xrightarrow{1} 2$ .

Способи (2) та (5) не максимізують кількість діамантів в першому коридорі. Серед способів (1), (3), та (4) тільки (3) максимізує кількість діамантів отриманих з другого коридору, це найкращий варіант для Петра. Зверніть уваго на те що спосіб (3) не максимізує кількість діамантів здобутих в 3 коридорі, ані сумарну кількість діамантів, але це єдина лексикографічно найбільша послідовність. Сумарна кількість діамантів яку Петро отримає це 22, це число потрібно повернути при такому виклику функції.



## Приклад градера

Формат вхідних даних наступний:

- рядок 1: три числа – значення  $N$ ,  $M$ , та  $K$ .



- рядки  $1 + i$ : три числа  $u[i]$ ,  $v[i]$ ,  $d[i]$  – описуючі коридор який починається в холі  $u[i]$  та закінчується в холі  $v[i]$  і має  $d[i]$  діамантів до видобудку.

Формат вихідних даних такий:

- рядок 1: одне число – значення повернуте викликом функції.