



關閉時間 (Closing Time)

匈牙利是個有 N 個城市的國家，編號由 0 至 $N - 1$ 。

這些城市被 $N - 1$ 條 **雙向** 道路連接，編號由 0 至 $N - 2$ 。對每個 j ，滿足 $0 \leq j \leq N - 2$ ，道路 j 連接城市 $U[j]$ 以及城市 $V[j]$ 且其長度為 $W[j]$ ，意即，耗費 $W[j]$ 單位時間可由此二城市之一通行至另一城市。每條道路連接兩個不同的城市，且每對城市被至多一條道路相連。

兩個不同城市 a 與 b 之間的 **路徑** (path) 是一個不同城市構成的序列 p_0, p_1, \dots, p_t ，滿足：

- $p_0 = a$,
- $p_t = b$,
- 對每個 i ($0 \leq i < t$)，有一條道路連接城市 p_i 與 p_{i+1} 。

從任一城市出發可經由若干道路通往任一其他城市，意即，任二不同城市間皆存在一路徑。我們可論證對任一對不同的城市此路徑唯一。

一條路徑 p_0, p_1, \dots, p_t 的 **長度** 為此路徑上連接相鄰城市的 t 條道路之長度之和。

在匈牙利，許多人會前往至兩個主要城市參加建國紀念日的慶祝活動。一旦慶典結束，他們就會回到各自的家。政府希望這些人群不要打擾到地方居民，所以他們計畫在某些時刻對出入城市進行管制。政府對每個城市指定了一個非負的 **關閉時間** (closing time)。政府決定所有關閉時間的總和不能大於 K 。更精確地說，對每個介於 0 與 $N - 1$ 之間的 i ，包含 0 與 $N - 1$ ，指定給城市 i 的關閉時間為一非負整數 $c[i]$ 。所有 $c[i]$ 的總和不大于 K 。

考慮一個城市 a 以及某個關閉時間的指派方式。我們稱城市 b 可由城市 a **抵達** 若且唯若 $b = a$ 或此二城市間的路徑 p_0, \dots, p_t (注意 $p_0 = a$ 且 $p_t = b$) 滿足下述條件：

- 路徑 p_0, p_1 的長度至多 $c[p_1]$ ，且
- 路徑 p_0, p_1, p_2 的長度至多 $c[p_2]$ ，且
- ...
- 路徑 $p_0, p_1, p_2, \dots, p_t$ 的長度至多 $c[p_t]$ 。

今年，兩個主要慶典的舉辦地在城市 X 與城市 Y 。對每種關閉時間的指定方式，其 **便利分數** 被定義為下列二數字之和：

- 從城市 X 出發可抵達的城市數量。
- 從城市 Y 出發可抵達的城市數量。

注意若從城市 X 與城市 Y 出發可抵達同一城市，則便利分數的計算中該城市會被計算 *兩次*。你的任務是計算所有可能的關閉時間指派方式中可達到的最大的便利分數。

實作細節 (Implementation Details)

你應實作下列程序。

```
int max_score(int N, int X, int Y, int64 K, int[] U, int[] V, int[] W)
```

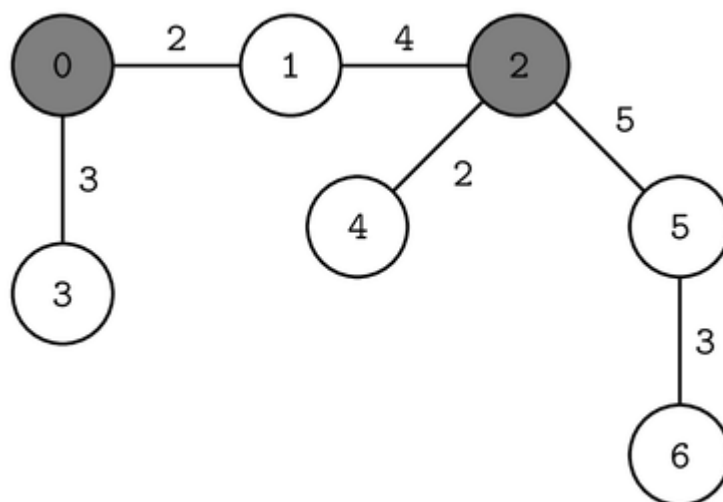
- N : 城市數量。
- X, Y : 主要慶典舉辦城市。
- K : 關閉時間總和上界。
- U, V : 長度為 $N - 1$ 的陣列，用來描述每條道路連接的城市。
- W : 長度為 $N - 1$ 的陣列，用來描述道路長度。
- 此程序應回傳所有可能的關閉時間指派方式中可達到的最大的便利分數。
- 對於每筆測資此程序可能被呼叫 **若干次**。

範例 (Example)

考慮下列呼叫：

```
max_score(7, 0, 2, 10,  
          [0, 0, 1, 2, 2, 5], [1, 3, 2, 4, 5, 6], [2, 3, 4, 2, 5, 3])
```

此例對應至下圖道路網：



假設關閉時間之指派方式如下：

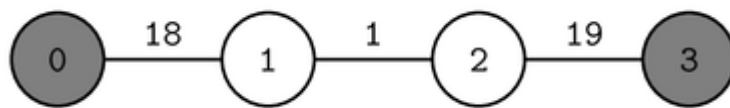
城市	0	1	2	3	4	5	6
關閉時間	0	4	0	3	2	0	0

注意所有關閉時間的總和為 9，並未超過 $K = 10$ 。城市 0、1 與 3 可由 X ($X = 0$) 出發抵達，而城市 1、2 與 4 可由城市 Y ($Y = 2$) 出發抵達。因此，此便利分數為 $3 + 3 = 6$ 。不存在關閉時間的指派方式能得到超過 6 的便利分數，故此程序應回傳 6。

同樣考慮下列呼叫：

```
max_score(4, 0, 3, 20, [0, 1, 2], [1, 2, 3], [18, 1, 19])
```

此例對應至下圖道路網：



假設關閉時間之指派方式如下：

城市	0	1	2	3
關閉時間	0	1	19	0

城市 0 可由城市 X ($X = 0$) 出發抵達，而城市 2 與 3 可由城市 Y ($Y = 3$) 出發抵達。因此，便利分數為 $1 + 2 = 3$ 。不存在關閉時間的指派方式能得到超過 3 的便利分數，故此程序應回傳 3。

限制 (Constraints)

- $2 \leq N \leq 200\,000$
- $0 \leq X < Y < N$
- $0 \leq K \leq 10^{18}$
- $0 \leq U[j] < V[j] < N$ (對每個 j 滿足 $0 \leq j \leq N - 2$)
- $1 \leq W[j] \leq 10^6$ (對每個 j 滿足 $0 \leq j \leq N - 2$)
- 從任一城市出發，可經由若干條道路通往任一其他城市。
- $S_N \leq 200\,000$ ，其中 S_N 為對於一筆測資之所有 `max_score` 的呼叫之 N 的總和。

子任務 (Subtasks)

若道路 i 連接城市 i 與 $i + 1$ (對每個 i 滿足 $0 \leq i \leq N - 2$)，我們稱該道路網為 **線性**。

1. (8 points) 城市 X 至城市 Y 的路經長度大於 $2K$ 。
2. (9 points) $S_N \leq 50$ ，道路網為線性。
3. (12 points) $S_N \leq 500$ ，道路網為線性。

4. (14 points) $S_N \leq 3\,000$ ，道路網為線性。
5. (9 points) $S_N \leq 20$
6. (11 points) $S_N \leq 100$
7. (10 points) $S_N \leq 500$
8. (10 points) $S_N \leq 3\,000$
9. (17 points) 無額外限制。

範例評分程式 (Sample Grader)

令 C 表示情境的個數，即，呼叫 `max_score` 的次數。此範例評分程式以下列格式讀取輸入：

- line 1: C

接下來為 C 個情境的描述。

範例評分程式以下列格式讀取各情境的描述：

- line 1: $N\ X\ Y\ K$
- line $2 + j\ (0 \leq j \leq N - 2)$: $U[j]\ V[j]\ W[j]$

對每一情境，範例評分程式以下列格式輸出一行：

- line 1: `max_score` 的回傳值