Zamykanie miast

Na Węgrzech jest N miast ponumerowanych od 0 do $N-1$.

Miasta są połączone siecią $N-1$ *dwukierunkowych* dróg ponumerowanych od 0 do $N-2$. Dla każdego j, takiego że $0 \leq j \leq N-2$, droga j łączy miasta $U[j]$ i $V[j]$ i ma długość $W[j]$, co oznacza, że przebycie tej drogi zajmuje $W[j]$ jednostek czasu. Każda droga łączy dwa różne miasta, a każda para miast jest połączona co najwyżej jedną drogą.

Ścieżką pomiędzy dwoma różnymi miastami a i b nazywamy ciąg $p_0, p_1, \ldots, p_t$ składający się z różnych miast, taki że:

- $p_0 = a$,
- $p_t = b$,
- dla każdego i ($0 \leq i < t$) istnieje droga łącząca miasta p_i i p_{i+1}.

Pomiędzy dowolną parą miast da się przejechać przy użyciu dróg. Innymi słowy, istnieje ścieżka pomiędzy dowolnymi dwoma miastami. Można udowodnić, że dla każdej pary miast istnieje dokładnie jedna ścieżka pomiędzy nimi.

Długością ścieżki $p_0, p_1, \ldots, p_t$ nazywamy sumę długości wszystkich t dróg łączących następujące po sobie miasta na ścieżce.

Wielu Węgrów wybiera się w podróż z okazji nadchodzącego święta narodowego, którego obchody odbędą się w dwóch miastach. Po zakończonych celebracjach udadzą się oni z powrotem do swoich domów. Węgierski rząd chce uchronić społeczności lokalne od związanych z tym niedogodności, dlatego planuje zamknąć wszystkie miasta. W tym celu każdemu miastu przypisany zostanie **czas zamknięcia**. Rząd zdecydował, że suma wszystkich czasów zamknięcia nie może przekroczyć K. Formalnie, dla każdego i pomiędzy 0 a $N-1$ włącznie, czas zamknięcia miasta i to nieujemna liczba całkowita $c[i]$. Suma wszystkich $c[i]$ nie może być większa niż K.

Rozważmy miasto a i pewne przypisanie czasów zamknięcia. Mówimy, że miasto b jest **osiągalne** z miasta a wtedy i tylko wtedy, gdy $b = a$ lub ścieżka $p_0, \ldots, p_t$ pomiędzy tymi miastami (a zatem w szczególności spełniająca $p_0 = a$ i $p_t = b$) spełnia następujące warunki:

- długość ścieżki p_0, p_1 wynosi co najwyżej $c[p_1]$ i
- długość ścieżki p_0, p_1, p_2 wynosi co najwyżej $c[p_2]$ i
- $\ldots$
- długość ścieżki $p_0, p_1, p_2, \ldots, p_t$ wynosi co najwyżej $c[p_t]$.

W tym roku obchody święta odbędą się w miastach X i Y. Dla każdego przypisania czasów zamknięcia **współczynnik wygody** zdefiniowany jest jako suma następujących dwóch liczb:

- liczby miast osiągalnych z miasta X oraz
- liczby miast osiągalnych z miasta Y.

Zauważmy, że jeśli miasto jest osiągalne zarówno z miasta X jak i z miasta Y, liczy się ono *dwukrotnie* przy obliczaniu współczynnika wygody.

Twoje zadanie polega na obliczeniu maksymalnego współczynnika wygody, jaki można osiągnąć przy pewnym przypisaniu czasów zamknięcia.

Szczegóły implementacji

Zaimplementuj poniższą procedurę.

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

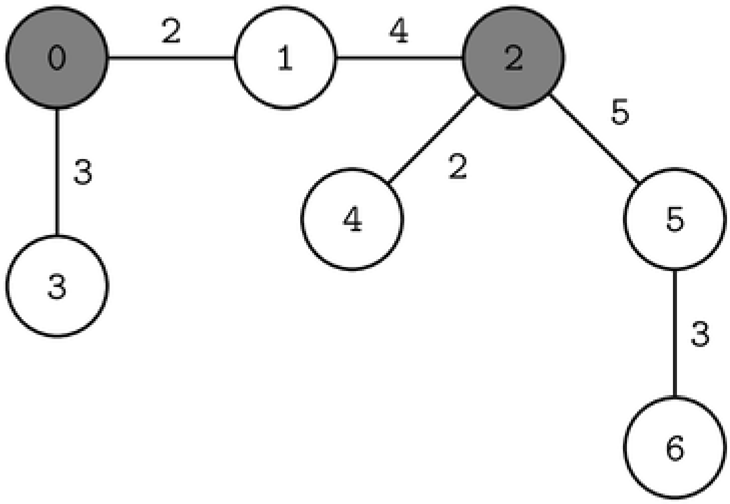
- N: liczba miast.
- X, Y: miasta w których odbywają się obchody.
- K: ograniczenie górne na sumę czasów zamknięcia.
- U, V: tablice długości $N-1$ opisujące drogi.
- W: tablica długości $N-1$ opisująca długości dróg.
- Ta procedura powinna zwrócić maksymalny współczynnik wygody, jaki można osiągnąć przy pewnym przypisaniu czasów zamknięcia.
- Ta procedura może być wywołana **wielokrotnie** w każdym przypadku testowym.

Przykład

Rozważmy poniższe wywołanie:

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

Odpowiada ono następującej sieci drogowej:



Przyjmijmy, że czasy zamknięcia są przypisane w następujący sposób:

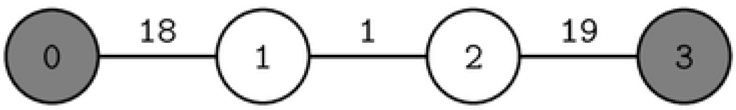
Miasto	0	1	2	3	4	5	6
Czas zamknięcia	0	4	0	3	2	0	0

Zauważmy, że suma wszystkich czasów zamknięcia to 9, czyli nie więcej niż $K = 10$. Miasta 0, 1, i 3 są osiągalne z miasta X ($X = 0$), a miasta 1, 2, i 4 są osiągalne z miasta Y ($Y = 2$). Zatem współczynnik wygody to $3 + 3 = 6$. Nie istnieje przypisanie czasów zamknięcia ze współczynnikiem wygody większym niż 6 i dlatego procedura powinna zwrócić 6.

Rozważmy teraz następujące wywołanie:

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

Odpowiada ono następującej sieci drogowej:



Przyjmijmy, że czasy zamknięcia są przypisane w następujący sposób:

Miasto	0	1	2	3
Czas zamknięcia	0	1	19	0

Miasto 0 jest osiągalne z miasta X ($X = 0$), a miasta 2 i 3 są osiągalne z miasta Y ($Y = 3$). Zatem współczynnik wygody to $1 + 2 = 3$. Nie istnieje przypisanie czasów zamknięcia ze współczynnikiem wygody większym niż 3 i dlatego procedura powinna zwrócić 3.

Ograniczenia

- $2 \leq N \leq 200\,000$
- $0 \leq X < Y < N$
- $0 \leq K \leq 10^{18}$
- $0 \leq U[j] < V[j] < N$ (dla każdego j, takiego że $0 \leq j \leq N-2$)
- $1 \leq W[j] \leq 10^6$ (dla każdego j, takiego że $0 \leq j \leq N-2$)
- Pomiędzy każdą parą miast da się przejechać przy użyciu dróg.
- $S_N \leq 200\,000$, gdzie S_N to suma argumentów N we wszystkich wywołaniach `max_score`.

Podzadania

Powiemy, że sieć drogowa jest **prosta** jeśli droga i łączy miasta i oraz $i+1$ (dla każdego i, takiego że $0 \leq i \leq N-2$).

1. (8 punktów) Długość ścieżki pomiędzy miastami X i Y jest większa niż $2K$.
2. (9 punktów) $S_N \leq 50$, sieć drogowa jest prosta.
3. (12 punktów) $S_N \leq 500$, sieć drogowa jest prosta.
4. (14 punktów) $S_N \leq 3\,000$, sieć drogowa jest prosta.
5. (9 punktów) $S_N \leq 20$
6. (11 punktów) $S_N \leq 100$
7. (10 punktów) $S_N \leq 500$
8. (10 punktów) $S_N \leq 3\,000$
9. (17 punktów) Brak dodatkowych ograniczeń.

Przykładowy program sprawdzający

Niech C oznacza liczbę scenariuszy testowych, czyli liczbę wywołań `max_score`. Format wejścia przykładowego programu sprawdzającego wygląda następująco:

- wiersz 1: C

Po nim następują opisy C scenariuszy testowych. Każdy z tych opisów ma następujący format:

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

Przykładowy program sprawdzający wypisuje pojedynczy wiersz dla każdego scenariusza testowego, w następującym formacie:

- wiersz 1: wartość zwrócona przez `max_score`