



Karol Pomorski to znany kierowca wyścigowy (zakrawający o korsarza drogowego) i nauczyciel informatyki (w tej kolejności). Wybiera się na wycieczkę po Wrocławiu.

Ponieważ stan dróg we Wrocławiu jest prawie tak tragiczny jak w Krakowie, używalne drogi możemy zamodelować jako drzewo $n - 1$ dróg łączących n skrzyżowań. Dzięki wysiłkom prezydenta Jacka Sutryka z każdego skrzyżowania we Wrocławiu da się dojechać do każdego innego drogami. Ponieważ Karol Pomorski porusza się z prędkościami łamiącymi prawa ruchu drogowego i fizyki, samochody, którymi jeździ trzeba często zmieniać, a także palić ogromne ilości paliwa.

Karol chce odwiedzić w pewnej kolejności wszystkie skrzyżowania. Przy każdym skrzyżowaniu poza ostatnim jeden z jego fanów wypożyczy mu swój samochód. Następnie przejedzie tym samochodem do kolejnego planowanego skrzyżowania do odwiedzenia, **nie odwiedzając** skrzyżowań po drodze. Ponieważ tak piłuje silnik, w i -tej podróży będzie musiał tankować co k_i skrzyżowań. Karol musi też zatankować w mieście początkowym i docelowym. Aby nie zostać na lodzie z pustym bakiem, każdy z fanów zaaranżował tak, żeby liczba dróg w danej podróży była podzielna przez k_i . Ponieważ nie każda stacja jest tak samo daleko od głównej drogi, na różnych stacjach zatankowanie baku kosztuje różną kwotę – c_i przy i -tym skrzyżowaniu.

Po pracy w Kuklu Karol jest dość bogaty, ale chciałby poprawnie zaksięgować swoją przejażdżkę. Pomóż mu policzyć sumaryczny koszt tankowania podczas każdego przejazdu.

Wejście

W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n ($2 \leq n \leq 50\,000$), oznaczająca liczbę skrzyżowań we Wrocławiu. Skrzyżowania są numerowane od 1 do n . W kolejnym wierszu znajduje się ciąg n liczb całkowitych c_i ($1 \leq c_i \leq 10\,000$) pooddzielanych pojedynczymi odstępami, oznaczających ceny całego baku paliwa przy poszczególnych skrzyżowaniach.

W kolejnych $n - 1$ wierszach znajdują się po dwie liczby całkowite a_i, b_i ($1 \leq a_i, b_i \leq n$), oznaczające, że skrzyżowania o numerach a_i i b_i łączy droga.

W następnym wierszu znajduje się ciąg n liczb całkowitych t_i , opisujący kolejność, w jakiej Karol Pomorski ma zamiar odwiedzić skrzyżowania (każda z liczb od 1 do n pojawi się w tym ciągu dokładnie raz).

Ostatni wiersz wejścia zawiera ciąg liczb całkowitych k_i oznaczających, co ile skrzyżowań Karol Pomorski musi tankować w i -tym przejeździe. Możesz założyć, że k_i zawsze dzieli odległość między skrzyżowaniami t_i a t_{i+1} .

Wyjście

Na wyjściu powinno znaleźć się $n - 1$ wierszy. W i -tym z nich powinna znaleźć się jedna liczba całkowita, oznaczająca koszt przejazdu między skrzyżowaniem o numerze t_i a t_{i+1} .

Przykład

Wejście dla testu pom0:

```
5
1 2 3 4 5
1 2
2 3
3 4
3 5
4 1 5 2 3
1 3 1 1
```

Wyjście dla testu pom0:

```
10
6
10
5
```



Ocenianie

To jest Kolano. Jeżeli Twój program zmieści się w limicie czasu (wynoszącym 2 sekundy) oraz pamięci i odpowie poprawnie, liczba punktów będzie proporcjonalna do szczytowego zużycia pamięci operacyjnej programu (im mniej tym lepiej). W przeciwnym wypadku, program otrzyma ∞ punktów. Granularność pomiaru pamięci może być narzucona przez ograniczenia systemu operacyjnego.