

გზების დახურვა

ქალაქ სურაბაიაში არის N ცალი კვანძი გადანომრილი 0-დან $(N - 1)$ -მდე. კვანძები დაკავშირებულია $N - 1$ ცალი ორმხრივი გზით, გადანომრილი 0-დან $(N - 2)$ -მდე ისე, რომ ნებისმიერი ორ კვანძი დაკავშირებულია უნიკალური მარშრუტით. i -ური გზა ($0 \leq i \leq N - 2$) აკავშირებს $U[i]$ და $V[i]$ კვანძებს.

გარემოს დაცვაზე ცნობიერების ასამაღლებლად, პაკ დენგკლეკი, როგორც სურაბაიას მერი, გეგმავს უმანქანო დღე მოაწესოს. ღონისძიების წასახალისებლად, პაკ დენგკლეკი ორგანიზებას გაუკეთებს გზების დახურვას. პაკ დენგკლეკი ჯერ აირჩევს არაუარყოფით მთელ რიცხვ k -ს, შემდეგ დახურავს ზოგიერთ გზას ისე, რომ თითოეული კვანძი უშუალოდ დაკავშირებული იქნება მაქსიმუმ k გზასთან, რომელიც არ არის დახურული. i -ური გზის დახურვის ფასია $W[i]$.

დაეხმარეთ პაკ დენგკლეკს იპოვოს მინიმალური ჯამური ფასი, რომელიც საჭიროა გზების დასახურად თითოეული არაუარყოფითი მთელი k -სთვის ($0 \leq k \leq N - 1$).

იმპლემენტაციის დეტალები

თქვენ უნდა მოახდინოთ შემდეგი ფუნქციის იმპლემენტაცია.

```
int64[] minimum_closure_costs(int N, int[] U, int[] V, int[] W)
```

- N : კვანძების რაოდენობა.
- U და V : $N - 1$ სიგრძის მასივი, სადაც $U[i]$ და $V[i]$ კვანძები დაკავშირებულია i გზით.
- W : $N - 1$ სიგრძის მასივი, სადაც $W[i]$ წარმოადგენს i გზის დახურვის ფასს.
- ამ ფუნქციამ უნდა დააბრუნოს N სიგრძის მასივი. ყოველი k ($0 \leq k \leq N - 1$), სადაც k -ური ელემენტი არის გზების დახურვის მინიმალური ჯამური ფასი, რომლის შემდეგაც თითოეული კვანძი დაკავშირებულია არაუმეტეს k დაუხურავ გზასთან.
- ეს ფუნქცია გამოიძახება მხოლოდ ერთხელ.

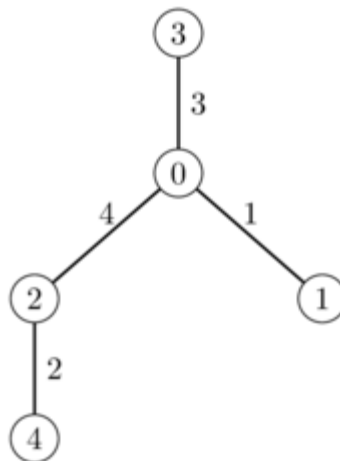
მაგალითები

მაგალითი 1

განვიხილოთ შემდეგი გამოძახება:

```
minimum_closure_costs(5, [0, 0, 0, 2], [1, 2, 3, 4], [1, 4, 3, 2])
```

ეს ნიშნავს, რომ მოცემულია 5 კვანძი და 4 გზა, რომლებიც აერთებენ შემდეგ კვანძთა წყვილებს (0, 1), (0, 2), (0, 3) და (2, 4). ამ გზების დახურვის ფასებია შესაბამისად 1, 4, 3 და 2.



მინიმალური ფასების მისაღებად:

- თუ $k = 0$, პაკ დენგკლეკმა უნდა დახუროს ყველა გზა და ჯამური ფასი იქნება $1 + 4 + 3 + 2 = 10$;
- თუ $k = 1$, უნდა დაიხუროს გზა 0 და გზა 1 ჯამური ფასით $1 + 4 = 5$;
- თუ $k = 2$, უნდა დაიხუროს გზა 0 ჯამური ფასით 1;
- თუ $k = 3$ ან $k = 4$, არცერთი გზის დახურვა საჭირო არ არის.

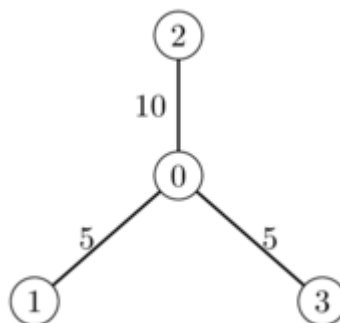
შესაბამისად, ფუნქცია `minimum_closure_costs` უნდა დააბრუნოს $[10, 5, 1, 0, 0]$.

მაგალითი 2

განვიხილოთ შემდეგი გამოცხება:

```
minimum_closure_costs(4, [0, 2, 0], [1, 0, 3], [5, 10, 5])
```

ეს ნიშნავს, რომ მოცემულია 4 კვანძი და 3 გზა, რომლებიც აერთებენ შემდეგ კვანძთა წყვილებს (0, 1), (2, 0), and (0, 3) ამ გზების დახურვის ფასებია შესაბამისად 5, 10 და 5.



To obtain the minimum costs:

- თუ $k = 0$, უნდა დაიხუროს ყველა გზა. ჯამური ფასი: $5 + 10 + 5 = 20$;
- თუ $k = 1$, უნდა დაიხუროს გზა 0 და გზა 2. ჯამური ფასი: $5 + 5 = 10$;
- თუ $k = 2$, უნდა დაიხუროს გზა 0 ან გზა 2. ფასი: 5;
- თუ $k = 3$, არცერთი გზის დახურვა საჭირო არ არის.

შესაბამისად, ფუნქციამ `minimum_closure_costs` უნდა დააბრუნოს $[20, 10, 5, 0]$.

შეზღუდვები

- $2 \leq N \leq 100\,000$
- $0 \leq U[i], V[i] \leq N - 1$ (ყველა $0 \leq i \leq N - 2$)
- არსებობს გზა კვანძთა ნებისმიერ წყვილისათვის .
- $1 \leq W[i] \leq 10^9$ (ყველა $0 \leq i \leq N - 2$)

ქვეამოცანები

1. (5 ქულა) $U[i] = 0$ (ყველა $0 \leq i \leq N - 2$)
2. (7 ქულა) $U[i] = i, V[i] = i + 1$ (ყველა $0 \leq i \leq N - 2$)
3. (14 ქულა) $N \leq 200$
4. (10 ქულა) $N \leq 2000$
5. (17 ქულა) $W[i] = 1$ (ყველა $0 \leq i \leq N - 2$)
6. (25 ქულა) $W[i] \leq 10$ (ყველა $0 \leq i \leq N - 2$)
7. (22 ქულა) დამატებითი შეზღუდვების გარეშე.

სანიმუშო გრაფერი

სანიმუშო გრაფერი კითხულობს მონაცემებს შემდეგი ფორმატით:

- სტრიქონი 1: N
- სტრიქონი $2 + i$ ($0 \leq i \leq N - 2$): $U[i] \ V[i] \ W[i]$

სანიმუშო გრაფერს ერთადერთ სტრიქონში გამოაქვს მონაცემები, რომელსაც აბრუნებს: `minimum_closure_costs`.