



Stansiyalar (stations)

Sinqapur İnternet Baş şəbəkəsi (SİB) 0-dan $(n - 1)$ -ə qədər **indekslərlə** nömrələnmiş n stansiyadan ibarətdir. Əlavə olaraq, 0-dan $(n - 2)$ -yə qədər nömrələnmiş ikitərəfli linklər var. Hər bir link iki fərqli stansiyanı birləşdirir. Bir linklə birləşmiş iki stansiya qonşu adlanır.

x stansiyasından y stansiyasına yol bir-birindən fərqli elə a_0, a_1, \dots, a_p stansiyaları ardıcılığıdır ki, $a_0 = x$, $a_p = y$ olsun, və yoldakı hər bir ardıcıl iki stansiya bir-birilə qonşudurlar. İstənilən x stansiyasından başqa bir y stansiyasına **düz bir ədəd** yol mövcuddur.

İstənilən x stansiyası paket (məlumat) yarada və paketin **hədəfi** adlanan istənilən başqa y stansiyasına göndərə bilər. Bu paket x -dən y -ə olan yeganə yol üzrə belə göndərilməlidir. Tutaq ki, hədəfi y stansiyası olan paket indi z stansiyasındadır ($z \neq y$). Bu halda, z stansiyası:

1. z -dən y -ə olan yeganə yoldakı z -in qonşusunu tapan **yönləndirmə proseduru**nu yerinə yetirir, və
2. bu paketi həmin qonşuya yönləndirir.

Lakin, stansiyaların yaddaşı kiçikdir və SİB-nin bütün linklərinin siyahısını yönləndirmə prosedurunda istifadə etmək üçün yaddaşa saxlamır.

Sizin tapşırığınız SİB üçün iki prosedurdan ibarət olan yönləndirməni yerinə yetirməkdir.

- İlk prosedura n , SİB-dəki linklərin siyahısı və $k \geq n - 1$ ədədi giriş kimi verilir. O, hər bir stansiyaya 0 və k arasında (hər ikisi daxil) müxtəlif **etiketlər** verir.
- İkinci prosedur etiketlər veriləndən sonra bütün stansiyalara yazılan yönləndirmə prosedurudur. Bu prosedurda **yalnız** bu giriş verilənləri mövcuddur:
 - s , hal-hazırda paketin olduğu stansiyanın **etiketi**,
 - t , paketin hədəfinin **etiketi** ($t \neq s$),
 - c , s stansiyasının bütün qonşularının **etiketlərinin** siyahısı.

Bu prosedur paketin yönləndirilməli olduğu qonşunun **etiketini** geri qaytarmalıdır.

Bir alt tapşırıqda, sizin həlliniz qiyməti stansiyalara verdiyiniz maksimum etiketin dəyərindən asılıdır (ümumiyyətlə, kiçik daha yaxşıdır).

İmplementasiya Detalları

Aşağıdakı prosedurları yerinə yetirməlisiniz:

```
int[] label(int n, int k, int[] u, int[] v)
```

- n : SİB-dəki stansiyaların sayı.
- k : istifadə edilə bilən maksimal etiket.
- u və v : linkləri göstərən $n - 1$ ölçülü massiv. Hər bir i üçün ($0 \leq i \leq n - 2$), i -ci link $u[i]$ və $v[i]$ indeksli stansiyaları birləşdirir.
- Bu prosedur yeganə n ölçülü L massivini qaytarmalıdır. Hər bir i üçün ($0 \leq i \leq n - 1$) $L[i]$ i indeksli stansiyaya verilmiş etiketdir. L massivinin bütün elementləri fərqli və 0 və k arasında (hər ikisi daxil) olmalıdır.

```
int find_next_station(int s, int t, int[] c)
```

- s : paketin olduğu stansiyanın etiketi.
- t : paketin hədəfinin etiketi.
- c : s -in bütün qonşularının etiketlərini göstərən massiv. c massivi artan sırada sıralanıb.
- Bu prosedur paketin yönləndirilməli olduğu s stansiyasının qonşusunun etiketini geri verməlidir.

Hər bir test bir və ya bir neçə asılı olmayan ssenarilərdən ibarətdir (məsələn, fərqli SİB verilənləri). r ssenarili test üçün, yuxardakı prosedurları çağıran **program** düz iki dəfə belə yerinə yetirilir.

İlk yerinə yetirmədə:

- `label` proseduru r dəfə çağırılır,
- geri qayıdan etiketlər qiymətləndirmə sistemi tərəfindən saxlanılır, və
- `find_next_station` çağrılmaz.

İkinci dəfə yerinə yetirmədə:

- `find_next_station` bir neçə dəfə çağrıla bilər. Hər çağrıda, **istənilən** ssenari seçilir, və həmin ssenaridə `label` prosedurundan qayıdan etiketlər `find_next_station` girişində istifadə olunur.
- `label` çağrılmaz.

Ümumi olaraq, ilk çağrıda statik və global dəyişənlərə yazılan məlumatlar `find_next_station` proseduru zamanı istifadə üçün mövcud deyil.

Nümunələr

Aşağıdakı çağırışa baxaq:

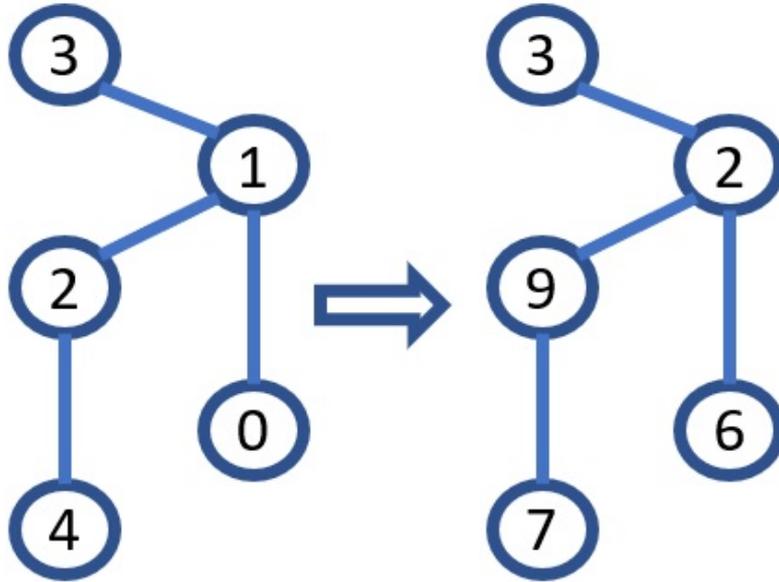
```
label(5, 10, [0, 1, 1, 2], [1, 2, 3, 4])
```

Toplmda 5 stansiya və $(0, 1)$, $(1, 2)$, $(1, 3)$ və $(2, 4)$ cütlərini birləşdirən 4 link var. Hər bir etiket 0 -dan $k = 10$ -a kimi bir tam ədəd ola bilər.

Aşağıdaki etiketleməni bildirmək üçün:

İndeks	Etiket
0	6
1	2
2	9
3	3
4	7

label proseduru [6, 2, 9, 3, 7] qaytarmalıdır. Aşağıdaki şəkildə ədədlər indeksləri (sol tərəfdəki paneldə) və təyin edilmiş etikətləri (sağ tərəfdəki paneldə) göstərir.



Fərz edin ki, etikətlər yuxarıda göstərilədiyi kimi təyin edilmişdir. Bu zaman aşağıda verilən çağırışı nəzərdən keçirin:

```
find_next_station(9, 6, [2, 7])
```

Bu o deməkdir ki, paket olan stansiyanın etiketi 9-dur və hədəf stansiyanın etiketi 6-dır. Hədəf stansiya gedən yolda olan stansiyaların etikətləri [9, 2, 6]-dir. Beləliklə funksiyadan nəticə olaraq 2 qaytarmalıdır.

Başqa mümkün çağırışı nəzərdən keçirin:

```
find_next_station(2, 3, [3, 6, 9])
```

Etiketi 3 olan hədəf stansiya etiketi 2 olan stansiyanın qonşusu olduğu üçün prosedur 3 qaytarmalıdır.

Limitlər

- $1 \leq r \leq 10$

Hər bir `label` çağırısı üçün:

- $2 \leq n \leq 1000$
- $k \geq n - 1$
- $0 \leq u[i], v[i] \leq n - 1$ (for all $0 \leq i \leq n - 2$)

Hər bir `find_next_station` çağırısı üçün giriş verilənləri istənilən bir öncəki `label` çağırısından götürülə bilər. Onun verdiyi etikətlərə baxaq. Onda:

- s və t iki fərqli stansiyanın etikətləridir.
- c s etikətli stansiyanın qonşularının etikətlərinin artan sırada ardıcılığıdır.

Hər bir test üçün, `find_next_station` proseduruna verilən bütün c massivlərin cəmi uzunluğu bütün ssenarilər üçün 100 000-i keçmir.

Alt Tapşırıqlar

1. (5 bal) $k = 1000$, heç bir stansiyanın 2-dən çox qonşusu yoxdur.
2. (8 bal) $k = 1000$, i -ci link $i + 1$ -ci və $\lfloor \frac{i}{2} \rfloor$ -ci stansiyaları birləşdirir.
3. (16 bal) $k = 1\,000\,000$, ən çoxu bir stansiyanın 2-dən çox qonşusu var.
4. (10 bal) $n \leq 8$, $k = 10^9$
5. (61 bal) $k = 10^9$

5-ci alt tapşırıqda qismi bal ala bilərsiniz. Qoy m bütün ssenarilər üçün `label` prosedurundan geri qayıdan maksimum etiket olsun. Sizin bu alt tapşırıq üçün balınız bu cədvələ əsasən hesablanacaq:

Maksimum Etiket	Bal
$m \geq 10^9$	0
$2000 \leq m < 10^9$	$50 \cdot \log_{5 \cdot 10^5} \left(\frac{10^9}{m} \right)$
$1000 < m < 2000$	50
$m \leq 1000$	61

Nümunə Grader (Qiymətləndirici)

Nümunə grader girişi bu formatda oxuyur:

- sətir 1: r

Növbəti r blok bir ssenarini təsvir edir. Hər bir blok bu formatdadır:

- sətir 1: n k
- sətir $2 + i$ ($0 \leq i \leq n - 2$): $u[i]$ $v[i]$
- sətir $1 + n$: q : find_next_station-a olan çağrılarının sayı.
- sətir $2 + n + j$ ($0 \leq j \leq q - 1$): $z[j]$ $y[j]$ $w[j]$: j -ci find_next_station çağrısında olan stansiyaların **indeksləri**. $z[j]$ stansiyasında paket var, $y[j]$ paketin hədəfidir, və $w[j]$ stansiyası paketin yönləndirilməli olduğu stansiyadır.

Nümunə grader bu formatda çıxışa verir:

- sətir 1: m

Bundan sonra girişdəki ardıcıl ssenarilərə uyğun r blok verilir. Hər bir blok bu formatdadır:

- sətir $1 + j$ ($0 \leq j \leq q - 1$): Bu ssenari üçün, j -ci find_next_station çağrısından qayıdan **etiketin** verildiği stansiyasının **indeksi**.

Nəzərə alın ki, sample grader hər yerinə yetirilmədən label və find_next_station prosedurlarının hər ikisini çağırır.