

Ključ

Arhitekt Timothy je pripravil novo igro pobega. V tej igri je n sob, oštevilčenih od 0 do $n - 1$. Od začetka ima vsaka soba natanko en ključ. Vsak ključ ima tip, ki je celo število med 0 in $n - 1$, vključujoče. Tip ključa v sobi i ($0 \leq i \leq n - 1$) je $r[i]$. Opazimo, da ima lahko več sob ključ istega tipa, t.j. vrednosti $r[i]$ niso nujno različne.

Obstaja tudi m **dvosmernih** konektorjev v igri, oštevilčenih od 0 do $m - 1$. Konektor j ($0 \leq j \leq m - 1$) povezuje par različnih sob $u[j]$ in $v[j]$. Dve sobi sta lahko povezani z več konektorji.

Igro igra en igralec, tako da zbira ključke in se premika med sobami s prečkanjem konektorjev. Rečemo, da je igralec **prečkal** konektor j , ko uporabi ta konektor za premik iz sobe $u[j]$ v sobo $v[j]$, ali obratno. Igralec lahko prečka konektor j , če je pred tem pobral ključ tipa $c[j]$.

V vsaki točki med igro se igralec nahaja v neki sobi x in lahko izvede eno izmed dveh tipov akcij:

- pobere ključ v sobi x , čigar tip je $r[x]$ (razen če ga je že pobral),
- prečka konektor j , kjer je bodisi $u[j] = x$ bodisi $v[j] = x$, če je igralec pred tem že pobral ključ tipa $c[j]$. Opazimo da igralec **nikoli** ne zavrže pobranega ključa.

Igralec **prične** igro v neki sobi s in nima nobenega ključa. Soba t je **dostopna** iz sobe s , če igralec, ki igro prične v sobi s , lahko izvede neko zaporedje zgornjih akcij, tako da se znajde v sobi t .

Za vsako sobo i ($0 \leq i \leq n - 1$), označimo število sob, ki jih je mogoče doseči iz sobe i , s $p[i]$. Timothy bi rad znal množico oznak i , ki dosežejo minimalno vrednost $p[i]$, za $0 \leq i \leq n - 1$.

Podrobnosti implementacije

Implementiraj naslednjo funkcijo:

```
int[] find_reachable(int[] r, int[] u, int[] v, int[] c)
```

- r : polje dolžine n . Za vsak i ($0 \leq i \leq n - 1$), je ključ sobe i tipa $r[i]$.
- u, v : dve polji dolžine m . Za vsak j ($0 \leq j \leq m - 1$), konektor j povezuje sobi $u[j]$ in $v[j]$.
- c : polje dolžine m . Za vsak j ($0 \leq j \leq m - 1$), tip ključa, potreben za prečkanje konektorja j , je $c[j]$.
- Funkcija naj vrne polje s dolžine n . Za vsak $0 \leq i \leq n - 1$, naj bo vrednost $s[i]$ enaka 1, če za vsak j , $0 \leq j \leq n - 1$, velja $p[i] \leq p[j]$. Sicer naj bo vrednost $s[i]$ enaka 0.

Primeri

1. primer

Razmisli o naslednjem klicu:

```
find_reachable([0, 1, 1, 2],  
               [0, 0, 1, 1, 3], [1, 2, 2, 3, 1], [0, 0, 1, 0, 2])
```

Če igralec igro prične v sobi 0, lahko izvede naslednje zaporedje akcij:

Trenutna soba	Akcija
0	Pobere ključ tipa 0
0	Prečka konektor 0 proti sobi 1
1	Pobere ključ tipa 1
1	Prečka konektor 2 proti sobi 2
2	Prečka konektor 2 proti sobi 1
1	Prečka konektor 3 proti sobi 3

Torej je soba 3 dostopna iz sobe 0. Podobno lahko sestavimo zaporedja, ki pokažejo, da so vse sobe dostopne iz sobe 0, iz česar sledi $p[0] = 4$. Spodnja tabela prikazuje katere sobe so dostopne, za vse začetne sobe:

Začetna soba i	Dostopne sobe	$p[i]$
0	[0, 1, 2, 3]	4
1	[1, 2]	2
2	[1, 2]	2
3	[1, 2, 3]	3

Najmanjša vrednost $p[i]$ čez vse sobe je 2, in je dosežena pri $i = 1$ in $i = 2$. Zatorej, funkcija vrne [0, 1, 1, 0].

2. primer

```
find_reachable([0, 1, 1, 2, 2, 1, 2],  
               [0, 0, 1, 1, 2, 3, 3, 4, 4, 5],  
               [1, 2, 2, 3, 3, 4, 5, 5, 6, 6],  
               [0, 0, 1, 0, 0, 1, 2, 0, 2, 1])
```

Spodnja tabela prikazuje dostopne sobe:

Začetna soba i	Dostopne sobe	$p[i]$
0	[0, 1, 2, 3, 4, 5, 6]	7
1	[1, 2]	2
2	[1, 2]	2
3	[3, 4, 5, 6]	4
4	[4, 6]	2
5	[3, 4, 5, 6]	4
6	[4, 6]	2

Najmanjša vrednost $p[i]$ čez vse sobe je 2, in je dosežena pri $i \in \{1, 2, 4, 6\}$. Zatorej, funkcija vrne [0, 1, 1, 0, 1, 0, 1].

3. primer

```
find_reachable([0, 0, 0], [0], [1], [0])
```

Spodnja tabela prikazuje dostopne sobe:

Začetna soba i	Dostopne sobe	$p[i]$
0	[0, 1]	2
1	[0, 1]	2
2	[2]	1

Najmanjša vrednost $p[i]$ čez vse sobe je 1, in je dosežena pri $i = 2$. Zatorej, funkcija vrne [0, 0, 1].

Omejitve

- $2 \leq n \leq 300\,000$
- $1 \leq m \leq 300\,000$
- $0 \leq r[i] \leq n - 1$ za vse $0 \leq i \leq n - 1$
- $0 \leq u[j], v[j] \leq n - 1$ in $u[j] \neq v[j]$ za vse $0 \leq j \leq m - 1$
- $0 \leq c[j] \leq n - 1$ za vse $0 \leq j \leq m - 1$

Podnaloge

1. (9 točk) $c[j] = 0$ za vse $0 \leq j \leq m - 1$ in $n, m \leq 200$
2. (11 točk) $n, m \leq 200$
3. (17 točk) $n, m \leq 2000$
4. (30 točk) $c[j] \leq 29$ (za vse $0 \leq j \leq m - 1$) in $r[i] \leq 29$ (za vse $0 \leq i \leq n - 1$)
5. (33 točk) Ni dodatnih omejitev.

Vzorčni ocenjevalnik

Vzorčni ocenjevalnik bere vhod v naslednji obliki:

- 1.vrstica: $n \ m$
- 2.vrstica: $r[0] \ r[1] \ \dots \ r[n-1]$
- $3 + j.$ ($0 \leq j \leq m-1$) vrstica: $u[j] \ v[j] \ c[j]$

Vzorčni ocenjevalnik izpiše rezultat klica `find_reachable` v naslednji obliki:

- 1. vrstica: $s[0] \ s[1] \ \dots \ s[n-1]$