

# Keys

Arhitekta Timoti je uključen u dizajn nove igre. U igri postoji  $n$  soba označenih brojevima od 0 do  $n - 1$ . Na početku, svaka soba sadrži tačno jedan ključ. Svaki ključ ima svoj tip, takođe označen brojevima od 0 do  $n - 1$ , inkluzivno. Tip ključa u sobi  $i$  ( $0 \leq i \leq n - 1$ ) je  $r[i]$ . Obratite pažnju da više soba može sadržati ključeve istog tipa tj. vrijednosti  $r[i]$  ne moraju biti različite.

Postoji  $m$  **dvosmjernih** konektora, označenih brojevima od 0 do  $m - 1$ . Konektor  $j$  ( $0 \leq j \leq m - 1$ ) povezuje par različitih soba  $u[j]$  i  $v[j]$ . Par soba može biti povezan pomoću više konektora.

Ovo je igra za jednog igrača, koji sakuplja ključeve kretanjem kroz sobe i premješta se iz sobe u sobu pomoću konektora.

Kažemo da igrač **prolazi** kroz konektor  $j$  ako koristi za prelazak iz sobe  $u[j]$  u sobu  $v[j]$  ili obrnuto. Igrač može proći kroz konektor  $j$  ako je ranije prikupio ključ tipa  $c[j]$ .

U svakom trenutku igre, igrač je u nekoj sobi  $x$  i ima na raspolaganju dva tipa akcija:

- pokupi ključ tipa  $r[x]$  iz sobe  $x$  (osim ako ga već nije pokupio),
- prolazi kroz konektor  $j$ , gdje je ili  $u[j] = x$  ili  $v[j] = x$ , pod uslovom da je ranije pokupio ključ tipa  $c[j]$ . Obratite pažnju da igrač **nikada** ne odbacuje već prikupljene ključeve.

Igrač **počinje** igru u nekoj sobi  $s$ , pri čemu nema nijedan ključ. Soba  $t$  je **dohvatljiva** iz sobe  $s$ , ako igrač koji počne igru u sobi  $s$  može izvesti niz akcija koje će ga dovesti u sobu  $t$ .

Za svaku sobu  $i$  ( $0 \leq i \leq n - 1$ ), označimo sa  $p[i]$  broj soba dohvatljivih iz sobe  $i$ . Timoti želi da odredi skup indeksa  $i$  koji dostižu minimalnu vrijednost  $p[i]$  nad svim  $0 \leq i \leq n - 1$ .

## Detalji implementacije

Potrebno je implementirati sljedeću funkciju:

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

- $r$ : niz dužine  $n$ . Za sve  $i$  ( $0 \leq i \leq n - 1$ ), ključ u sobi  $i$  ima tip  $r[i]$ .
- $u, v$ : dva niza dužine  $m$ . Za sve  $j$  ( $0 \leq j \leq m - 1$ ), konektor  $j$  povezuje sobe  $u[j]$  i  $v[j]$ .
- $c$ : niz dužine  $m$ . Za sve  $j$  ( $0 \leq j \leq m - 1$ ),  $c[j]$  je tip ključa potreban za prolaz kroz konektor  $j$ .
- Ova funkcija vraća niz  $a$  dužine  $n$ . Za svako  $0 \leq i \leq n - 1$ , vrijednost  $a[i]$  treba da je 1 ako za sve  $j$  takve da  $0 \leq j \leq n - 1$ ,  $p[i] \leq p[j]$ . U suprotnom, vrijednost  $a[i]$  treba da je 0.

# Primjeri

## Primjer 1

Razmotrimo sljedeći poziv funkcije:

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

Ako igrač počne igru u sobi 0, može izvesti sljedeći niz akcija:

| Trenutna soba | Akcija                         |
|---------------|--------------------------------|
| 0             | Pokupi ključ tipa 0            |
| 0             | Prolazi konektorom 0 do sobe 1 |
| 1             | Pokupi ključ tipa 1            |
| 1             | Prolazi konektorom 2 do sobe 2 |
| 2             | Prolazi konektorom 2 do sobe 1 |
| 1             | Prolazi konektorom 3 do sobe 3 |

Dakle, soba 3 je dohvatljiva iz sobe 0. Slično, možemo konstruisati nizove akcija koji će pokazati da su sve sobe dohvatljive iz sobe 0, pa zaključujemo da je  $p[0] = 4$ . Tabela prikazuje za svaku početnu sobu odgovarajuću listu dohvatljivih soba:

| Početna soba $i$ | Dohvatljive sobe | $p[i]$ |
|------------------|------------------|--------|
| 0                | [0, 1, 2, 3]     | 4      |
| 1                | [1, 2]           | 2      |
| 2                | [1, 2]           | 2      |
| 3                | [1, 2, 3]        | 3      |

Najmanja vrijednost  $p[i]$  je 2 i ona se dostiže za  $i = 1$  ili  $i = 2$ . Dakle, funkcija vraća [0, 1, 1, 0].

## Primjer 2

```
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])
```

Tabela prikazuje za svaku početnu sobu odgovarajuću listu dohvatljivih soba:

| Početna soba $i$ | Dohvatljive 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      |

Najmanja vrijednost  $p[i]$  je 2 i dostiže se za  $i \in \{1, 2, 4, 6\}$ . Dakle, funkcija vraća [0, 1, 1, 0, 1, 0, 1].

### Primjer 3

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

Tabela prikazuje za svaku početnu sobu odgovarajuću listu dohvatljivih soba:

| Početna soba $i$ | Dohvatljive sobe | $p[i]$ |
|------------------|------------------|--------|
| 0                | [0, 1]           | 2      |
| 1                | [0, 1]           | 2      |
| 2                | [2]              | 1      |

Najmanja vrijednost  $p[i]$  je 1 i dostiže se kada je  $i = 2$ . Funkcija vraća [0, 0, 1].

## Ograničenja

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

## Podzadaci

1. (9 bodova)  $c[j] = 0$  za sve  $0 \leq j \leq m - 1$  i  $n, m \leq 200$
2. (11 bodova)  $n, m \leq 200$
3. (17 bodova)  $n, m \leq 2000$
4. (30 bodova)  $c[j] \leq 29$  (za sve  $0 \leq j \leq m - 1$ ) i  $r[i] \leq 29$  (za sve  $0 \leq i \leq n - 1$ )

5. (33 bodova) Nema dodatnih ograničenja.

## Program za ocjenjivanje (Sample Grader)

Format za učitavanje podataka je:

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

Program za ocjenjivanje štampa rezultat pozivanja `find_reachable` u sljedećem formatu:

- red 1:  $a[0] \ a[1] \ \dots \ a[n-1]$