

Ключі

Архітектор Тимофій спроектував нову квест-кімнату. Гра складається з n кімнат пронумерованих від 0 до $n - 1$. На початку кожна кімната містить рівно один ключ. Кожен ключ відноситься до певного типу, що є цілим числом від 0 до $n - 1$ включно. Тип ключа в кімнаті i ($0 \leq i \leq n - 1$) дорівнює $r[i]$. Зауважте, що кілька кімнат може містити ключ одного типу, тобто величини $r[i]$ не обов'язково різні.

В грі також присутні m **двосторонніх** коридорів, пронумерованих від 0 до $m - 1$. Коридор j ($0 \leq j \leq m - 1$) з'єднує пару різних кімнат $u[j]$ та $v[j]$. Пара кімнат може бути зв'язана кількома коридорами.

В гру грає один гравець, який збирає ключі та рухається між кімнатами, проходячи коридорами. Кажемо, що гравець **проходить** через коридор j , коли він використовує його для руху з кімнати $u[j]$ до кімнати $v[j]$, або навпаки. Гравець може використати коридор j лише в тому випадку, коли він до цього забрав ключ типу $c[j]$.

В будь-який момент гри гравець знаходиться у кімнаті x і може вчинити одну із 2 дій:

- забрати ключ у кімнаті x , тип якого $r[x]$ (якщо він його ще не отримав),
- пройти через коридор j , у якого або $u[j] = x$, або $v[j] = x$, за умови що він вже має ключ типу $c[j]$. Зауважимо, що гравець **ніколи** не втрачає ключ, який вже отримав.

Гравець **розпочинає** гру у кімнаті s без жодного ключа. Кімната t **досяжна** з кімнати s , якщо гравець, розпочинаючи гру у кімнаті s , може здійснити певну послідовність дій, описаних вище, і опинитись у кімнаті t .

Для кожної кімнати i ($0 \leq i \leq n - 1$) позначимо кількість кімнат досяжних із i як $p[i]$. Тимофій хотів би знайти множину індексів i , що відповідає мінімальному значенню $p[i]$ серед всіх $0 \leq i \leq n - 1$.

Деталі реалізації

Вам потрібно реалізувати наступну процедуру:

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

- r : масив довжини n . Для кожного i ($0 \leq i \leq n - 1$), ключ в кімнаті i має тип $r[i]$.
- u, v : два масива довжини m , для яких j ($0 \leq j \leq m - 1$), коридор j поєднує кімнати $u[j]$ та $v[j]$.
- c : масив довжини m . Для кожного j ($0 \leq j \leq m - 1$) тип ключа необхідного щоб пройти через коридор j є $c[j]$.

- Процедура повинна повернути масив a довжини n . Для кожного $0 \leq i \leq n - 1$, величина $a[i]$ має дорівнювати 1, якщо для всіх j $0 \leq j \leq n - 1$, $p[i] \leq p[j]$. В іншому випадку $a[i]$ має дорівнювати 0.

Приклади

Приклад 1

Розглянемо наступний виклик:

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

Якщо гравець розпочинає гру у кімнаті 0, він може здійснити наступну послідовність дій:

Поточна кімната	дія
0	Забрати ключ типу 0
0	Пройти коридором 0 до кімнати 1
1	Забрати ключ типу 1
1	Пройти коридором 2 до кімнати 2
2	Пройти коридором 2 до кімнати 1
1	Пройти коридором 3 до кімнати 3

Отже, кімната 3 досяжна із кімнати 0. Аналогічно можна побудувати конструкції, які продемонструють, що всі кімнати досяжні із кімнати 0, отже $p[0] = 4$. Табличка нижче показує досяжні кімнати для всіх початкових:

Початкова кімната i	Досяжні кімнати	$p[i]$
0	[0, 1, 2, 3]	4
1	[1, 2]	2
2	[1, 2]	2
3	[1, 2, 3]	3

Найменше значення $p[i]$ серед всіх кімнат 2, і воно досягається при $i = 1$ та $i = 2$. Отже процедура повинна повернути [0, 1, 1, 0].

Приклад 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])
```

Табличка нижче показує досяжні кімнати для всіх початкових:

Початкова кімната i	досяжні кімнати	$p[i]$
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

Найменше значення $p[i]$ серед всіх кімнат 2, і воно досягається при $i \in \{1, 2, 4, 6\}$. Отже процедура повинна повернути [0, 1, 1, 0, 1, 0, 1].

Приклад 3

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

Табличка нижче показує досяжні кімнати для всіх початкових:

Початкова кімната i	досяжні кімнати	$p[i]$
0	[0, 1]	2
1	[0, 1]	2
2	[2]	1

Найменше значення $p[i]$ серед всіх кімнат 1 і воно досягається при $i = 2$. Отже процедура повинна повернути [0, 0, 1].

Обмеження

- $2 \leq n \leq 300\,000$
- $1 \leq m \leq 300\,000$
- $0 \leq r[i] \leq n - 1$ для всіх $0 \leq i \leq n - 1$
- $0 \leq u[j], v[j] \leq n - 1$ and $u[j] \neq v[j]$ для всіх $0 \leq j \leq m - 1$

- $0 \leq c[j] \leq n - 1$ для всіх $0 \leq j \leq m - 1$

Підзадачі

1. (9 балів) $c[j] = 0$ для всіх $0 \leq j \leq m - 1$ і $n, m \leq 200$
2. (11 балів) $n, m \leq 200$
3. (17 балів) $n, m \leq 2000$
4. (30 балів) $c[j] \leq 29$ (для всіх $0 \leq j \leq m - 1$) і $r[i] \leq 29$ (для всіх $0 \leq i \leq n - 1$)
5. (33 бали) без додаткових обмежень.

Приклад модуля перевірки

Приклад модуль перевірки зчитує дані у наступному форматі:

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

Приклад модуль перевірки виводить значення, що повернула процедура `find_reachable`, у наступному форматі:

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