

מפתחות

טימותי הארכיטקט תכנן משחק בריחה חדש. במשחק הזה, ישנם n חדרים ממוספרים מ-0 עד $n-1$. בתחילת המשחק כל חדר מכיל בדיוק מפתח אחד. לכל מפתח יש סוג, שהוא מספר שלם בין 0 ל- $n-1$. כולל. הסוג של המפתח בחדר i ($0 \leq i \leq n-1$) הוא $r[i]$. שימו לב שמספר חדרים יכולים להכיל מפתחות מאותו הסוג, כלומר הערכים $r[i]$ אינם בהכרח שונים.

בנוסף, ישנם m מסדרונות דו-כיווניים במשחק, ממוספרים מ-0 עד $m-1$. המסדרון ה- j ($0 \leq j \leq m-1$) מחבר זוג חדרים שונים $u[j]$ ו- $v[j]$. זוג חדרים יכול להיות מחובר ע"י מספר מסדרונות.

במשחק משתתף שחקן יחיד שאוסף מפתחות ועובר בין חדרים באמצעות המסדרונות. אנו אומרים שהשחקן עובר במסדרון ה- j כשהוא משתמש במסדרון הזה כדי להגיע מהחדר $u[j]$ אל החדר $v[j]$ או להפך. השחקן יכול לעבור במסדרון ה- j רק אם הוא אסף מפתח מסוג $c[j]$ לפני כן.

בכל שלב במשחק, השחקן נמצא בחדר כלשהו x ויכול לבצע שני סוגים של פעולות:

- לאסוף את המפתח בחדר x , שהסוג שלו הוא $r[x]$ (אלא אם כן הוא כבר אסף אותו),
- לעבור במסדרון j , כאשר $u[j] = x$ או $v[j] = x$, אם השחקן אסף מפתח מסוג $c[j]$ לפני כן. שימו לב שהשחקן לעולם לא משליך מפתח שהוא אסף.

השחקן מתחיל את המשחק בחדר כלשהו s , כשאין לו אף מפתח. חדר t נקרא נגיש מחדר s אם השחקן שמתחיל את המשחק בחדר s יכול לבצע סדרה כלשהי של פעולות כמתואר לעיל, ולהגיע לחדר t .

לכל חדר i ($0 \leq i \leq n-1$) נסמן את מספר החדרים הנגישים מחדר i ב- $p[i]$. טימותי מעוניין למצוא את קבוצת האינדקסים i שעבורם הערך של $p[i]$ מינימלי, מבין $0 \leq i \leq n-1$.

פרטי מימוש

עליכם לממש את הפונקציה הבאה:

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

- r : מערך באורך n . לכל i ($0 \leq i \leq n-1$), המפתח בחדר i הוא מסוג $r[i]$.
- u, v : שני מערכים באורך m . לכל j ($0 \leq j \leq m-1$), המסדרון j מחבר את החדרים $u[j]$ ו- $v[j]$.
- c : מערך באורך m . לכל j ($0 \leq j \leq m-1$), סוג המפתח שדרוש כדי לעבור במסדרון j הוא $c[j]$.
- הפונקציה הזו צריכה להחזיר מערך a באורך n . לכל i ($0 \leq i \leq n-1$), הערך של $a[i]$ צריך להיות 1 אם לכל j כך ש- $0 \leq j \leq n-1$, $p[i] \leq p[j]$. אחרת, הערך של $a[i]$ צריך להיות 0.

דוגמאות

דוגמה 1

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

אם השחקן מתחיל את המשחק בחדר 0, הוא יכול לבצע את סדרת הפעולות הבאה:

פעולה	חדר נוכחי
אסוף מפתח מסוג 0	0
עבור במסדרון 0 לחדר 1	0
אסוף מפתח מסוג 1	1
עבור במסדרון 2 לחדר 2	1
עבור במסדרון 2 לחדר 1	2
עבור במסדרון 3 לחדר 3	1

לפיכך חדר 3 נגיש מחדר 0. באופן דומה, אנו יכולים לבנות סדרות שיראו שכל החדרים נגישים מחדר 0, ומכך נובע כי $p[0] = 4$. הטבלה הבאה מראה את החדרים הנגישים מכל חדרי ההתחלה.

i	החדרים הנגישים	החדר ההתחלתי i
0	[0, 1, 2, 3]	0
1	[1, 2]	1
2	[1, 2]	2
3	[1, 2, 3]	3

הערך הקטן ביותר של $p[i]$ מבין כל החדרים הוא 2, והוא מתקבל עבור $i = 1$ ו- $i = 2$. לפיכך, הפונקציה צריכה להחזיר [0, 1, 1, 0].

דוגמה 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])
```

הטבלה הבאה מראה את החדרים הנגישים:

i	החדרים הנגשים	$p[i]$
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

הערך הקטן ביותר של $p[i]$ מבין כל החדרים הוא 2, והוא מתקבל עבור $i \in \{1, 2, 4, 6\}$. לפיכך, הפונקציה צריכה להחזיר $[0, 1, 1, 0, 1, 0, 1]$.

דוגמה 3

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

הטבלה הבאה מראה את החדרים הנגשים:

i	החדרים הנגשים	$p[i]$
0	[0, 1]	2
1	[0, 1]	2
2	[2]	1

הערך הקטן ביותר של $p[i]$ מבין כל החדרים הוא 1, והוא מתקבל עבור $i = 2$. לפיכך, הפונקציה צריכה להחזיר $[0, 0, 1]$.

מגבלות

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

תת משימות

1. (9 נקודות) $c[j] = 0$ לכל $0 \leq j \leq m - 1$ ו- $n, m \leq 200$
2. (11 נקודות) $n, m \leq 200$
3. (17 נקודות) $n, m \leq 2000$

4. (30 נקודות) $c[j] \leq 29$ (לכל $0 \leq j \leq m - 1$) ו- $r[i] \leq 29$ (לכל $0 \leq i \leq n - 1$)
5. (33 נקודות) ללא מגבלות נוספות.

גריידר לדוגמה

הגריידר לדוגמה קורא את הקלט בפורמט הבא:

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

הגריידר לדוגמה מדפיס את ערך ההחזרה של `find_reachable` בפורמט הבא:

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