

# Prepájanie superstromov (supertrees)

V Singapore sa nachádza veľký prírodný park Gardens by the Bay. V tomto parku stojí  $n$  obrovských veží, ktorým miestni hovoria superstromy. Superstromy sú očíslované od 0 po  $n - 1$ . Chceli by sme ich prepojiť. Postavíme preto **nula alebo viac** mostov. Každý most prepojí dva rôzne superstromy. Každá dvojica superstromov bude priamo prepojená nanajvýš jedným mostom. Po každom moste sa bude dať chodiť oboma smermi.

Cesta z veže  $x$  na vežu  $y$  je postupnosť jednej alebo viacerých veží, v ktorej platí:

- postupnosť začína vežou  $x$ ,
- postupnosť končí vežou  $y$ ,
- všetky veže v postupnosti sú **navzájom rôzne**,
- každé dve po sebe idúce veže sú prepojené mostom.

Všimnite si, že z tejto definície vyplýva, že pre každú vežu  $i$  existuje práve jedna cesta z  $i$  na  $i$ . Tiež si všimnite, že počet ciest z veže  $i$  na vežu  $j$  je nutne rovný počtu ciest z veže  $j$  na vežu  $i$ .

Hlavný staviteľ mostov by ich chcel postaviť tak, aby pre každé  $0 \leq i, j \leq n - 1$  platilo, že z veže  $i$  na vežu  $j$  vedie presne  $p[i][j]$  rôznych ciest. Je zaručené, že všetky tieto čísla sú z rozsahu  $0 \leq p[i][j] \leq 3$ .

Zistite, či sa tento cieľ dá dosiahnuť. Ak áno, nájdite jeden spôsob, ako postaviť mosty.

## Implementation details

Naprogramujte nasledujúcu funkciu:

```
int construct(int[][] p)
```

- $p$ : pole rozmerov  $n \times n$  obsahujúce plánované počty ciest.
- Túto funkciu grader zavolá práve raz.
- Ak existuje riešenie, vaša funkcia by mala práve raz zavolať funkciu `build` (viď popis nižšie) a nahlásiť tak, ktoré mosty postaviť. Následne má vaša funkcia vrátiť 1.
- Ak riešenie neexistuje, vaša funkcia nesmie zavolať `build` a musí vrátiť 0.

Ak riešenie existuje, to vaše odovzdáte volaním nasledujúcej funkcie gradera:

```
void build(int[][] b)
```

- $b$ : pole rozmerov  $n \times n$ , v ktorom  $b[i][j] = 1$  znamená, že veže  $i$  a  $j$  chceme prepojiť mostom a  $b[i][j] = 0$  znamená, že medzi nimi priamy most nebude.
- Nezabudnite, že toto pole musí spĺňať dve podmienky:  $b[i][j] = b[j][i]$  pre  $0 \leq i, j \leq n - 1$  a tiež  $b[i][i] = 0$  pre  $0 \leq i \leq n - 1$ .

## Examples

### Example 1

Grader zavolať vašu funkciu nasledovne:

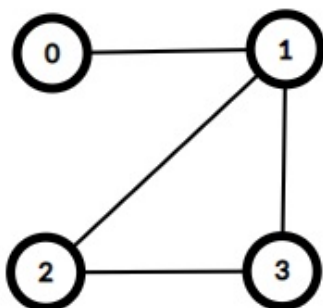
```
construct([[1, 1, 2, 2], [1, 1, 2, 2], [2, 2, 1, 2], [2, 2, 2, 1]])
```

Toto  $p$  hovorí, že z veže 0 na vežu 1 má viesť práve jedna cesta, a pre každú inú dvojicu veží  $(x, y)$  takú, že  $0 \leq x < y \leq 3$ , musia z  $x$  na  $y$  viesť práve dve cesty.

Toto sa dá dosiahnuť postavením štyroch mostov, ktoré prepoja nasledujúce dvojice veží:  $(0, 1)$ ,  $(1, 2)$ ,  $(1, 3)$  a  $(2, 3)$ .

Aby ste toto riešenie odovzdali graderu, musí vaša funkcia `construct` spraviť napríklad nasledovné volanie:

- `build([[0, 1, 0, 0], [1, 0, 1, 1], [0, 1, 0, 1], [0, 1, 1, 0]])`



Následne má vaša funkcia vrátiť hodnotu 1.

Existujú aj iné volania funkcie `build` ktoré by tiež boli akceptované.

### Example 2

Grader zavolať vašu funkciu nasledovne:

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

Tento vstup hovorí, že medzi vežami 0 a 1 nemá existovať žiadna cesta. Toto vieme dosiahnuť práve tak, že nepostavíme žiadne mosty. Vaša funkcia má teda zavolať

- `build([[0, 0], [0, 0]])`

a následne skončiť s návratovou hodnotou 1.

### Example 3

Grader zavolať vašu funkciu nasledovne:

```
construct([[1, 3], [3, 1]])
```

Jednou z požiadaviek uvedených v tomto vstupe je, že z veže 0 na vežu 1 majú viesť tri cesty. Toto nevieme dosiahnuť. Vaša funkcia preto nesmie spraviť žiadne volanie funkcie `build` a musí skončiť s návratovou hodnotou 0.

## Constraints

- $1 \leq n \leq 1000$
- $p[i][i] = 1$  (pre  $0 \leq i \leq n - 1$ )
- $p[i][j] = p[j][i]$  (pre  $0 \leq i, j \leq n - 1$ )
- $0 \leq p[i][j] \leq 3$  (pre  $0 \leq i, j \leq n - 1$ )

## Subtasks

1. (11 points)  $p[i][j] = 1$  (pre  $0 \leq i, j \leq n - 1$ )
2. (10 points)  $p[i][j] = 0$  alebo 1 (pre  $0 \leq i, j \leq n - 1$ )
3. (19 points)  $p[i][j] = 0$  alebo 2 (pre  $i \neq j, 0 \leq i, j \leq n - 1$ )
4. (35 points)  $0 \leq p[i][j] \leq 2$  (pre  $0 \leq i, j \leq n - 1$ ) a je zaručené, že nejaká taká sada mostov existuje
5. (21 points)  $0 \leq p[i][j] \leq 2$  (pre  $0 \leq i, j \leq n - 1$ )
6. (4 points) Bez ďalších obmedzení.

## Sample grader

Ukázkový grader očakáva vstup v nasledujúcom formáte:

- line 1:  $n$
- line  $2 + i$  ( $0 \leq i \leq n - 1$ ):  $p[i][0] \ p[i][1] \ \dots \ p[i][n - 1]$

Výstup z ukázkového gradera:

- line 1: návratová hodnota `construct`.
- line  $2 + i$  ( $0 \leq i \leq n - 1$ ):  $b[i][0] \ b[i][1] \ \dots \ b[i][n - 1]$
- Ak `construct` vráti 0, grader nevypíše  $b$ .