

Connecting Supertrees (supertrees)

Gardens by the Bay is een groot natuurpark in Singapore. In het park staan n torens, ook bekend als superbomen. Deze torens zijn gelabeld van 0 tot en met $n - 1$. We willen een verzameling maken van **nul of meer** bruggen. Elke brug verbindt twee verschillende torens en kan in **beide** richtingen worden gebruikt om over te steken. Twee bruggen mogen niet hetzelfde paar torens verbinden.

Een pad van toren x naar toren y is een reeks van één of meer torens waarvoor geldt:

- Het eerste element van de reeks is x
- Het laatste element van de reeks is y
- Alle elementen in de reeks zijn **verschillend**
- Elke twee opeenvolgende elementen (torens) in de reeks zijn verbonden met een brug

Merk op dat er per definitie precies één pad van een toren naar zichzelf bestaat en dat het aantal verschillende paden van toren i naar toren j hetzelfde is als het aantal verschillende paden van toren j naar toren i .

De hoofdarchitect, verantwoordelijk voor het ontwerp, wil de bruggen graag op zo'n manier bouwen dat er voor alle $0 \leq i, j \leq n - 1$ precies $p[i][j]$ verschillende paden zijn van toren i naar toren j , waarbij $0 \leq p[i][j] \leq 3$ geldt.

Maak een verzameling van bruggen die aan de eisen van de architect voldoet, of bepaal dat dit onmogelijk is.

Implementatiedetails

Je moet de volgende functie implementeren:

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

- p : een $n \times n$ array die de eisen van de architect vertegenwoordigt.
- Als een geldige constructie mogelijk is moet de functie precies één aanroep maken naar `build` (zie hieronder) om de constructie door te geven en daarna moet het 1 teruggeven.
- Anders moet de functie 0 teruggeven zonder enige aanroep naar `build`.
- Deze functie wordt precies één keer aangeroepen.

De functie `build` is als volgt gedefinieerd:

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

- b : een $n \times n$ array, met $b[i][j] = 1$ als er een brug bestaat die toren i en toren j verbindt, of anders $b[i][j] = 0$.
- Merk op dat de array moet voldoen aan de voorwaarde $b[i][j] = b[j][i]$ voor alle $0 \leq i, j \leq n - 1$ en $b[i][i] = 0$ voor alle $0 \leq i \leq n - 1$.

Voorbeelden

Voorbeeld 1

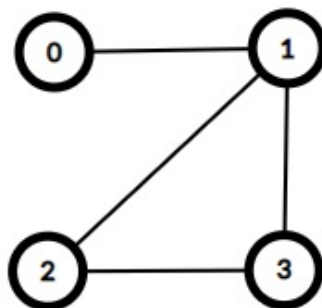
Neem de volgende aanroep:

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

Dit betekent dat er precies één pad moet bestaan van toren 0 naar toren 1. Voor elk ander paar torens (x, y) , waarbij $0 \leq x < y \leq 3$, moeten er precies twee paden zijn van toren x naar toren y . Dit kan worden bereikt met 4 bruggen, die de torens $(0, 1)$, $(1, 2)$, $(1, 3)$ en $(2, 3)$ verbinden.

Om deze oplossing door te geven moet de `construct` functie de volgende aanroep maken:

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



Daarna moet de functie 1 teruggeven.

In dit geval zijn er meerdere constructies mogelijk die aan de eisen voldoen, deze worden allemaal goedgekeurd.

Voorbeeld 2

Neem de volgende aanroep:

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

Dit betekent dat er geen pad moet zijn tussen de twee torens. Dit kan enkel worden bereikt door geen bruggen te hebben.

Daarom moet de `construct` functie de volgende aanroep maken:

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

Daarna moet de `construct` functie 1 teruggeven.

Voorbeeld 3

Neem de volgende aanroep:

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

Dit betekent dat er precies 3 paden van toren 0 naar toren 1 moeten zijn. Aan deze set van eisen kan niet worden voldaan. Daarom moet de `construct` functie 0 teruggeven zonder enige aanroep naar `build`.

Randvoorwaarden

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

Subtaken

1. (11 punten) $p[i][j] = 1$ (voor alle $0 \leq i, j \leq n - 1$)
2. (10 punten) $p[i][j] = 0$ of 1 (voor alle $0 \leq i, j \leq n - 1$)
3. (19 punten) $p[i][j] = 0$ of 2 (voor alle $i \neq j, 0 \leq i, j \leq n - 1$)
4. (35 punten) $0 \leq p[i][j] \leq 2$ (voor alle $0 \leq i, j \leq n - 1$) en er is precies één constructie die aan de voorwaarden voldoet.
5. (21 punten) $0 \leq p[i][j] \leq 2$ (voor alle $0 \leq i, j \leq n - 1$)
6. (4 punten) Geen aanvullende randvoorwaarden.

Voorbeeldgrader

De voorbeeldgrader leest de invoer in het volgende formaat:

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

De uitvoer van de voorbeeldgrader is in het volgende formaat:

- regel 1: de waarde teruggegeven door `construct`.

Als de waarde van `construct` 1 is, schrijft de voorbeeldgrader daarnaast ook weg:

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