

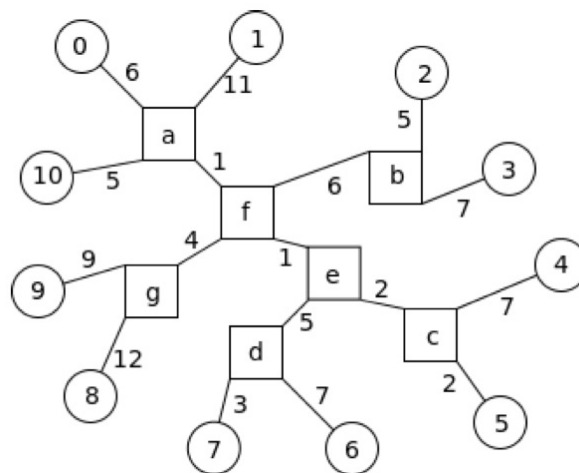
# Dorpen

Er zijn  $N$  kleine dorpen in Kazachstan, genummerd van  $0$  tot en met  $N - 1$ . Er is ook een onbekend aantal grote steden. De kleine dorpen en grote steden van Kazachstan noemen we allemaal *gemeenten*.

Alle gemeenten van Kazachstan zijn verbonden door één netwerk van bidirectionele snelwegen. Elke snelweg verbindt twee verschillende gemeenten, en elk paar gemeenten is met elkaar verbonden door ten hoogste één snelweg. Voor elk paar gemeenten  $a$  en  $b$  is er een unieke manier waarop men van  $a$  naar  $b$  kan gaan via de snelwegen, zolang geen snelweg meer dan éénmaal wordt gebruikt.

Het is geweten dat elk klein dorp direct verbonden is met één enkele andere gemeente, en dat elke grote stad direct verbonden is met drie of meer gemeenten.

De volgende afbeelding toont een netwerk van **11** kleine dorpen en **7** grote steden. Kleine dorpen worden weergegeven als cirkels en gelabeld met getallen, grote steden worden weergegeven als vierkanten en gelabeld met letters.



Elke snelweg heeft een positieve gehele lengte. De afstand tussen twee gemeenten is de kleinste som van lengtes van snelwegen die men moet afreizen om van de ene gemeente naar de andere te gaan.

Voor elke grote stad  $C$  kunnen we afstand  $r(C)$  meten tot het kleine dorp dat het verst verwijderd is van die stad. Een grote stad  $C$  is een *hub* als diens afstand  $r(C)$  de kleinste is van alle grote steden. De afstand tussen een hub en een klein dorp dat het verst verwijderd is van een hub zullen we  $R$  noemen. Dus,  $R$  is de kleinste van alle waarden  $r(C)$ .

In het voorbeeld hierboven is klein dorp **8** het verst verwijderde dorp van stad **a**, en de afstand tussen die twee is  $r(a) = 1 + 4 + 12 = 17$ . Ook voor stad **g** is  $r(g) = 17$ . (Eén van de kleine dorpen die het verst weg is van **g** is dorp 6.) De enige hub in het bovenstaande voorbeeld is stad **f**, met  $r(f) = 16$ . Dus, is het bovenstaande voorbeeld is  $R = 16$ .

Het verwijderen van een hub deelt het netwerk op in verschillende samenhangende delen. Een hub is *gebalanceerd* als elk van deze delen ten hoogste  $\lfloor N/2 \rfloor$  kleine dorpen bevat. (We benadrukken dat we de grote steden niet meetellen.) Met  $\lfloor x \rfloor$  bedoelen we het grootste geheel getal dat niet groter is dan  $x$ .

In ons voorbeeld is stad  $f$  een hub. Als we stad  $f$  verwijderen, valt het netwerk uiteen in 4 samenhangende delen. Deze vier delen bevatten de volgende sets van kleine dorpen:  $\{0, 1, 10\}$ ,  $\{2, 3\}$ ,  $\{4, 5, 6, 7\}$ , en  $\{8, 9\}$ . Geen van deze delen bevat meer dan  $\lfloor 11/2 \rfloor = 5$  dorpen, dus stad  $f$  is een gebalanceerde hub.

## Taak

In het begin is de enige informatie die je hebt over het netwerk van gemeenten en snelwegen, het aantal kleine dorpen  $N$ . Je kent het aantal grote steden niet. Je weet ook niets over de layout van de snelwegen in het land. Je kan enkel nieuwe informatie krijgen door vragen te stellen over de afstanden tussen paren van kleine dorpen.

Jouw taak is om te bepalen:

- In elke subtaak: de afstand  $R$ .
- In subtaken 3 tot 6: of het netwerk een gebalanceerde hub bevat.

Je moet de functie `hubDistance` implementeren. De grader zal verschillende testcases evalueren in één uitvoering. Het aantal testcases per uitvoering is maximaal **40**. Voor elke testcase zal de grader jouw functie `hubDistance` exact éénmaal aanroepen. Zorg dat jouw functie alle benodigde variabelen initialiseert elke keer dat ze wordt aangeroepen.

- `hubDistance(N, sub)`
  - $N$ : het aantal kleine dorpen.
  - `sub`: het nummer van de subtaak (uitgelegd in de paragraaf Subtaken).
  - Als `sub` 1 of 2 is, dan mag de functie  $R$  of  $-R$  teruggeven
  - Als `sub` groter is dan 2, dan moet de functie  $R$  teruggeven als er een gebalanceerde hub bestaat, en anders moet ze  $-R$  teruggeven.

Je functie `hubDistance` kan info over het snelwegennetwerk opvragen door de graderfunctie `getDistance(i, j)` aan te roepen. Deze functie geeft de afstand tussen kleine dorpen  $i$  en  $j$  terug. Let op dat als  $i$  en  $j$  gelijk zijn, de functie  $0$  teruggeeft. Ze geeft ook  $0$  terug als de argumenten ongeldig zijn.

## Subtaken

In elke testcase:

- is  $N$  tussen **6** en **110** (inclusief).
- is de afstand tussen elke twee verschillende kleine dorpen tussen 1 and 1,000,000 (inclusief).

Het aantal vragen dat je programma mag stellen is beperkt. De limiet varieert per subtaak, zoals gegeven in de tabel hieronder. Als je programma de limiet op het aantal vragen overschrijdt, wordt het beëindigd en geldt dit als een fout antwoord.

subtaak	punten	aantal vragen	vind gebalanceerde hub	bijkomende beperkingen
1	13	$\frac{N(N-1)}{2}$	NEE	geen
2	12	$\lceil \frac{7N}{2} \rceil$	NEE	geen
3	13	$\frac{N(N-1)}{2}$	JA	geen
4	10	$\lceil \frac{7N}{2} \rceil$	JA	elke grote stad is verbonden met <i>exact</i> drie gemeenten
5	13	$5N$	JA	geen
6	39	$\lceil \frac{7N}{2} \rceil$	JA	geen

Met  $\lceil x \rceil$  bedoelen we het kleinste geheel getal dat groter dan of gelijk is aan  $x$ .

### Voorbeeldgrader

Let op dat het subtaaknummer een deel is van de input. Het gedrag van de voorbeeldgrader verandert naargelang het subtaaknummer.

De voorbeeldgrader leest input van het bestand `towns.in` in het volgende formaat:

- lijn 1: Subtaaknummer en het aantal testcases.
- lijn 2:  $N_1$ , het aantal kleine dorpen in de eerste testcase.
- de volgende  $N_1$  lijnen: Het  $j$ -de getal ( $1 \leq j \leq N_1$ ) in de  $i$ -de van deze lijnen ( $1 \leq i \leq N_1$ ) is de afstand tussen kleine dorpen  $i - 1$  en  $j - 1$ .
- Daarop volgen de volgende testcases. Ze zijn gegeven in hetzelfde formaat als de eerste testcase.

Voor elke testcase print de voorbeeldgrader de returnwaarde van `hubDistance` en het aantal aanroepen, op verschillende lijnen.

Het inputbestand dat overeenkomt met het voorbeeld hierboven is:

```

1 1
11
0 17 18 20 17 12 20 16 23 20 11
17 0 23 25 22 17 25 21 28 25 16
18 23 0 12 21 16 24 20 27 24 17
20 25 12 0 23 18 26 22 29 26 19
17 22 21 23 0 9 21 17 26 23 16
12 17 16 18 9 0 16 12 21 18 11
20 25 24 26 21 16 0 10 29 26 19
16 21 20 22 17 12 10 0 25 22 15
23 28 27 29 26 21 29 25 0 21 22
20 25 24 26 23 18 26 22 21 0 19
11 16 17 19 16 11 19 15 22 19 0

```

Dit formaat is nogal verschillend van een gewone specificatie van een lijst snelwegen. Noteer dat het toegelaten is om de voorbeeldgraders te wijzigen zodanig dat ze een ander inputformaat gebruiken.