



Draugas

Šioje užduotyje konstruosime socialinį tinklą iš n žmonių, sunumeruotų nuo 0 iki $n - 1$. Kai kurios žmonių poros bus draugais. Žmogui x tapus žmogaus y draugu, y irgi taps x draugu.

Tinklas bus sukonstruotas per n žingsnių, irgi sunumeruotų nuo 0 iki $n - 1$. Žmogus, kurio numeris i , į tinklą įtraukiamas i -ojo žingsnio metu. 0-inio žingsnio metu į tinklą įtraukiamas 0-is žmogus ir jis tuo metu yra vienintelis žmogus tinkle. Per kiekvieną kitą žingsnį *vedėjas*, kuriuo gali būti bet kuris tuo metu tinkle jau esantis žmogus, į tinklą įtraukia vieną naują žmogų. Per i -ąjį žingsnį *vedėjas* įtraukia i -ąjį žmogų, naudodamas vieną iš trijų taisyklių:

- *IAmYourFriend* padaro i -ąjį žmogų *vedėjo* draugu.
- *MyFriendsAreYourFriends* padaro i -ąjį žmogų *visų* *vedėjo* tuo metu turimų draugų draugu. Atkreipkite dėmesį, kad ši taisyklė i -ojo žmogaus nepadaro *vedėjo* draugu.
- *WeAreYourFriends* padaro i -ąjį žmogų *visų* *vedėjo* tuo metu turimų draugų ir paties *vedėjo* draugu.

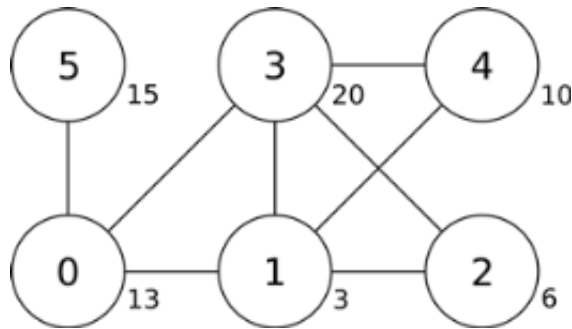
Sukonstravę tinklą turėsime išsirinkti kai kuriuos žmones apklausai. Pasirinktus žmones vadinsime *rinkiniu*. Kadangi draugai dažnai turi panašius pomėgius, rinkinyje negali būti nei vienos poros žmonių, kurie yra tarpusavyje draugai. Kiekvienam žmogui yra priskirtas *patikimumas*, išreikštas teigiamu sveikuoju skaičiumi, ir mes norime sudaryti rinkinį, kurio bendras patikimumas būtų pats didžiausias.

Pavyzdys

žingsnis	vedėjas	taisyklė	naujos draugų poros
1	0	IAmYourFriend	(1, 0)
2	0	MyFriendsAreYourFriends	(2, 1)
3	1	WeAreYourFriends	(3, 1), (3, 0), (3, 2)
4	2	MyFriendsAreYourFriends	(4, 1), (4, 3)
5	0	IAmYourFriend	(5, 0)

Iš pradžių tinkle yra vienintelis 0-asis žmogus. 1-ojo žingsnio metu *vedėjas* yra 0-asis žmogus ir jis įtraukia 1-ąjį žmogų naudodamas *IAmYourFriend* taisyklę, todėl jie tampa draugais. 2-ojo žingsnio metu *vedėjas* ir vėl yra 0-asis žmogus ir jis įtraukia 2-ąjį žmogų naudodamas *MyFriendsAreYourFriends* taisyklę, todėl 1-asis žmogus (vienintelis *vedėjo* draugas) ir 2-asis tampa draugais. 3-iojo žingsnio metu *vedėjas* yra 1-asis žmogus ir jis įtraukia 3-įjį naudodamas *WeAreYourFriends* taisyklę, todėl 3-iasis žmogus tampa 1-ojo (*vedėjo*), 0-ojo ir 2-ojo (*vedėjo* draugu) draugu. Lentelėje taip pat aprašyti 4-asis ir 5-asis žingsniai. Sukonstruotas tinklas pavaizduotas žemiau esančioje diagramoje, kurioje skritulių viduje esantys skaičiai žymi žmonių numerius, o skaičiai prie skritulių – patikimumą. Rinkinio, sudaryto iš 3-iojo ir 5-ojo žmogaus, bendras patikimumas būtų

$20 + 15 = 35$ ir tai yra didžiausias galimas patikimumas.



Užduotis

Turėdami visų žingsnių aprašymą ir visų žmonių patikimumo vertes, raskite rinkinį, kurio bendras patikimumas yra pats didžiausias. Tam parašykite funkciją `findSample`.

- `findSample(n, confidence, host, protocol)`
 - `n`: žmonių skaičius.
 - `confidence`: n ilgio masyvas; `confidence[i]` nurodo i -ojo žmogaus patikimumą.
 - `host`: n ilgio masyvas; `host[i]` nurodo i -ojo žingsnio vedėją.
 - `protocol`: n ilgio masyvas; `protocol[i]` nurodo, kuri taisyklė buvo naudojama per i -ąjį žingsnį ($0 < i < n$): 0, jei tai naudojama `IamYourFriend`, 1 jei `MyFriendsAreYourFriends`, ir 2 jei `WeAreYourFriends`.
 - Kadangi 0-ojoje dalyje nėra vedėjo, `host[0]` ir `protocol[0]` vertės nėra apibrėžtos, todėl jūsų programa neturėtų bandyti jų naudoti.
 - Funkcija turėtų gražinti didžiausią galimą rinkinio bendą patikimumą.

Dalinės užduotys

Kai kuriose dalinėse užduotyse vedėjai naudosis ne visomis taisyklėmis, kaip parašyta lentelėje žemiau.

dalinė užduotis	taškai	n	patikimumas	naudojamos taisyklės
1	11	$2 \leq n \leq 10$	$1 \leq \text{patikimumas} \leq 1\,000\,000$	Visos trys taisyklės.
2	8	$2 \leq n \leq 1\,000$	$1 \leq \text{patikimumas} \leq 1\,000\,000$	Tik <code>MyFriendsAreYourFriends</code> .
3	8	$2 \leq n \leq 1\,000$	$1 \leq \text{patikimumas} \leq 1\,000\,000$	Tik <code>WeAreYourFriends</code> .
4	19	$2 \leq n \leq 1\,000$	$1 \leq \text{patikimumas} \leq 1\,000\,000$	Tik <code>IAMYourFriend</code> .
5	23	$2 \leq n \leq 1\,000$	Visų žmonių patikimumas yra 1.	<code>MyFriendsAreYourFriends</code> ir <code>IAMYourFriend</code>
6	31	$2 \leq n \leq 100\,000$	$1 \leq \text{patikimumas} \leq 10\,000$	Visos trys taisyklės.

Reikalavimai

Pateikite vieną failą, pavadintą `friend.c`, `friend.cpp` arba `friend.pas`. Šiame faile pateikite parašytą funkciją `findSample`. Jei programuojate C/C++ kalba, įtraukite antraštinį failą `friend.h`.

C/C++ programa

```
int findSample(int n, int confidence[], int host[], int protocol[]);
```

Pascal programa

```
function findSample(n: longint, confidence: array of longint, host: array of longint; protocol: array of longint): longint;
```

Pavyzdinis vertintojas

Pavyzdinis vertintojas nuskaito duomenis tokiu formatu:

- 1-oji eilutė: `n`
- 2-oji eilutė: `confidence[0], ..., confidence[n-1]`
- 3-ioji eilutė: `host[1], protocol[1], host[2], protocol[2], ..., host[n-1], protocol[n-1]`

Pavyzdinis vertintojas išves funkcijos `findSample` grąžinamą vertę.