



## Prijatelj

Gradimo socialno omrežje z  $n$  ljudmi, oštevilčenimi  $0, \dots, n - 1$ . Nekateri pari ljudi v omrežju bodo prijatelji. Če oseba  $x$  postane prijatelj osebe  $y$ , potem je tudi  $y$  prijatelj osebe  $x$ .

Ljudi dodajamo v omrežje v  $n$  korakih, prav tako oštevilčenih od  $0$  do  $n - 1$ . V koraku  $i$  dodamo osebo  $i$ . V koraku  $0$  dodamo v omrežje osebo  $0$  kot prvo in edino osebo. V vsakem od naslednjih  $n - 1$  korakov ima ena oseba, ki je že v omrežju, status *gostitelja*. V koraku  $i$  ( $0 < i < n$ ) dotični gostitelj doda osebo  $i$  v omrežje z eno izmed sledečih operacij:

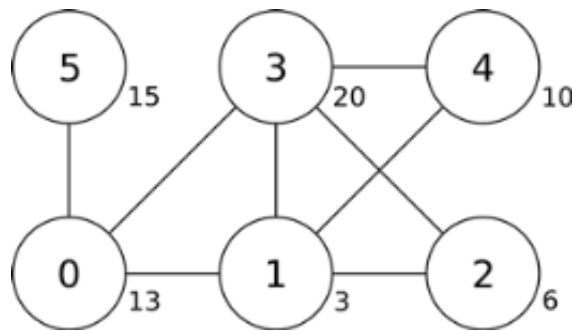
- *IAmYourFriend*: oseba  $i$  se spoprijatelji samo z gostiteljem.
- *MyFriendsAreYourFriends*: oseba  $i$  se spoprijatelji z *vsemi* osebami, ki so trenutno že prijatelji gostitelja. Ta operacija ne spoprijatelji osebe  $i$  z gostiteljem.
- *WeAreYourFriends*: oseba  $i$  se spoprijatelji tako z gostiteljem, kot tudi z *vsemi* osebami, ki so trenutno že prijatelji gostitelja.

Ko se gradnja omrežja zaključi, želimo iz njega izbrati *vzorec* oseb za anketiranje. Ker načeloma velja, da imajo prijatelji podobne interese, vzorec ne sme vsebovati nobenega para oseb, ki sta prijatelja. Vsaka oseba ima določeno mero *zaupanja*, ki je izražena s pozitivnim celim številom. Naša naloga je, da poiščemo vzorec z največjim možnim skupnim zaupanjem.

## Primer

korak	gostitelj	operacija	dodana prijateljstva
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)

Na začetku omrežje vsebuje zgolj osebo  $0$ . Gostitelj koraka  $1$  (oseba  $0$ ) povabi v omrežje osebo  $1$  z operacijo *IAmYourFriend*, torej ti dve osebi postaneta prijatelja. Gostitelj koraka  $2$  (ponovno oseba  $0$ ) povabi osebo  $2$  z *MyFriendsAreYourFriends*, kar spoprijatelji osebi  $1$  in  $2$ , saj je oseba  $1$  edini prijatelj osebe  $0$ . Gostitelj koraka  $3$  (oseba  $1$ ) doda osebo  $3$  z *WeAreYourFriends*, kar spoprijatelji osebo  $3$  z osebo  $1$  (gostiteljem) ter osebama  $0$  in  $2$  (prijatelja gostitelja). Koraka  $4$  in  $5$  sta prikazana v zgornji tabeli, končno omrežje pa na spodnji sliki. Številke v krogih označujejo osebe, povezave med krogi pa njihova prijateljstva. Poleg vsakega kroga je zapisana številka, ki podaja zaupanje dotične osebe. Vzorec, ki vsebuje osebi  $3$  in  $5$ , ima skupno zaupanje enako  $20 + 15 = 35$ , kar je tudi največje možno zaupanje tega omrežja.



## Naloga

Pri podanem zaporedju korakov gradnje omrežja, v končnem omrežju poišči vzorec z največjim možnim zaupanjem.

Implementirajte funkcijo `findSample`.

- `findSample(n, confidence, host, protocol)`
  - `n`: število ljudi.
  - `confidence`: tabela dolžine `n`; `confidence[i]` podaja zaupanje osebe `i`.
  - `host`: tabela dolžine `n`; `host[i]` podaja gostitelja koraka `i`.
  - `protocol`: tabela dolžine `n`; `protocol[i]` podaja oznako operacije uporabljene v koraku `i` ( $0 < i < n$ ): 0 za operacijo `IAmYourFriend`, 1 za operacijo `MyFriendsAreYourFriends` in 2 za operacijo `WeAreYourFriends`.
  - Ker v koraku 0 gostitelja ni, sta `host[0]` in `protocol[0]` nedefinirana, vaš program pa do teh dveh vrednosti ne sme dostopati.
  - Funkcija naj vrne največjo možno vrednost zaupanja v končnem omrežju.

## Podnaloge

Nekatere podnaloge uporabljajo zgolj podmnožico operacij, kot je prikazano v spodnji tabeli.

podnaloga	št. točk	$n$	zaupanje	uporabljene operacije
1	11	$2 \leq n \leq 10$	$1 \leq \text{zaupanje} \leq 1,000,000$	Vse tri operacije
2	8	$2 \leq n \leq 1,000$	$1 \leq \text{zaupanje} \leq 1,000,000$	Samo <code>MyFriendsAreYourFriends</code>
3	8	$2 \leq n \leq 1,000$	$1 \leq \text{zaupanje} \leq 1,000,000$	Samo <code>WeAreYourFriends</code>
4	19	$2 \leq n \leq 1,000$	$1 \leq \text{zaupanje} \leq 1,000,000$	Samo <code>IAmYourFriend</code>
5	23	$2 \leq n \leq 1,000$	Vse osebe imajo zaupanje 1	Samo <code>MyFriendsAreYourFriends</code> in <code>IAmYourFriend</code>
6	31	$2 \leq n \leq 100,000$	$1 \leq \text{confidence} \leq 10,000$	Vse tri operacije

## Podrobnosti implementacije

Oddati morate natanko eno datoteko poimenovano `friend.c`, `friend.cpp` ali `friend.pas`. V tej datoteki implementirajte funkcijo, ki je opisana zgoraj, imeti pa mora natakó tak podpis kot je podan spodaj. V jeziku C/C++ morate obvezno vključiti tudi header datoteko `friend.h`.

### Programski jezik C/C++

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

### Programski jezik Pascal

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

### Vzorčni ocenjevalnik

Vzorčni ocenjevalnik bere vhod v sledečem formatu:

- vrstica 1: `n`
- vrstica 2: `confidence[0], ..., confidence[n-1]`
- vrstica 3: `host[1], protocol[1], host[2], protocol[2], ..., host[n-1], protocol[n-1]`

Vzorčni ocenjevalnik bo izpisal vrednost, ki jo vrne funkcija `findSample`.