



Prijatelj

Gradimo socijalnu mrežu koja će se sastojati od n ljudi označenih brojevima $0, 1, \dots, n - 1$. Neki parovi ljudi u našoj socijalnoj mreži bit će prijatelji. Prijateljstva su obostrana. Dakle, ako je osoba x prijatelj osobe y onda je i osoba y prijatelj osobe x .

Socijalnu mrežu gradimo u n faza koje su također označene brojevima od 0 do $n - 1$. U i -toj fazi dodajemo osobu i u socijalnu mrežu. U fazi 0 dodajemo osobu 0 kao jedinu osobu u socijalnoj mreži. U svakoj od sljedećih $n - 1$ faza dodajemo novu osobu preko *domaćina* koji može biti bilo koja osoba koja je već dodana u socijalnu mrežu. U fazi i ($0 < i < n$) domaćin može dodati osobu i na neki od sljedeća tri načina:

- *JaSamTiFrend* -- osoba i sprijateljuje se samo s domaćinom
- *MojiFrendoviSuTiFrendovi* -- osoba i sprijateljuje se sa *svim* trenutnim prijateljima domaćina, ali ne i s domaćinom
- *MiSmoTiFrendovi* -- osoba i sprijateljuje se s domaćinom i svim njegovim trenutnim prijateljima

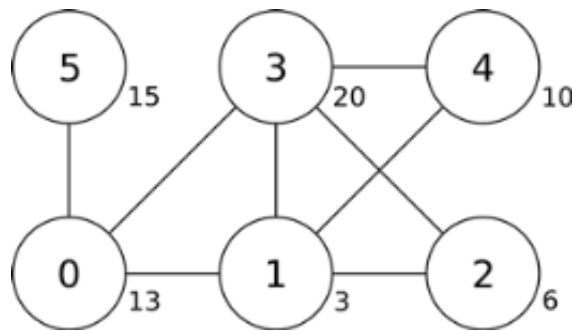
Nakon što izgradimo socijalnu mrežu htjeli bismo odabrati *uzorak* za naše socijalno istraživanje o utjecaju pandi na kvalitetu života čovjeka. Prijatelji obično imaju slične interese pa se u našem uzorku ne smiju nalaziti dva čovjeka koji su prijatelji. Također, svaka osoba ima *ljubav prema pandama*, izraženu kao pozitivan cijeli broj. Željeli bismo za naše istraživanje odabrati uzorak koji ima najveću moguću *ljubav prema pandama*.

Primjer

faza	domaćin	način dodavanja	dodana prijateljstva
1	0	<i>JaSamTiFrend</i>	(1, 0)
2	0	<i>MojiFrendoviSuTiFrendovi</i>	(2, 1)
3	1	<i>MiSmoTiFrendovi</i>	(3, 1), (3, 0), (3, 2)
4	2	<i>MojiFrendoviSuTiFrendovi</i>	(4, 1), (4, 3)
5	0	<i>JaSamTiFrend</i>	(5, 0)

Na samom početku u mreži je samo osoba 0. Domaćin faze 1 (osoba 0) dodaje osobu 1 *JaSamTiFrend* načinom pa oni postaju prijatelji. Domaćin faze 2 je opet osoba 0 i ona dodaje osobu 2 načinom *MojiFrendoviSuTiFrendovi* pa osoba 2 postaje prijatelj samo osobi 0. Domaćin faze 3, osoba 1, dodaje osobu 3 načinom *MiSmoTiFrendovi* pa osoba 3 prijateljuje s osobom 1 (domaćinom) i osobama 0 i 2 (prijateljima domaćina). Faze 4 i 5 također su prikazane u gornjoj tablici.

Završni izgled socijalne mreže prikazan je na idućoj slici (brojevi u krugovima su redni brojevi osoba, a brojevi pored krugova njihova *ljubav prema pandama*).



Uzmemo li osobe 3 i 5 u uzorak *ljubav prema pandama* iznosit će $20 + 15 = 35$, što je i maksimum.

Zadatak

Za zadani opis svake faze izgradnje socijalne mreže i *ljubavi prema pandama* svake osobe, nađite uzorak koji najviše *voli pande*. Implementirajte funkciju `findSample`.

- `findSample(n, confidence, host, protocol)`
 - `n`: broj ljudi.
 - `confidence`: niz duljine n ; `confidence[i]` je *ljubav prema pandama* osobe i .
 - `host`: niz duljine n ; `host[i]` je domaćin i -te faze.
 - `protocol`: niz duljine n ; `protocol[i]` označava način dodavanja u fazi i za ($0 < i < n$): 0 za *JaSamTiFrend*, 1 za *MojiFrendoviSuTiFrendovi*, and 2 za *MiSmoTiFrendovi*.
 - Budući da nema domaćina u fazi 0, nije određeno što su vrijednosti `host[0]` i `protocol[0]` pa ih vaš program ne bi trebao koristiti.
 - Funkcija treba vratiti maksimalnu moguću *ljubav prema pandama* nekog uzorka.

Podzadaci

Neki podzadaci ne koriste sve načine dodavanja.

podzadatak	broj bodova	n	<i>ljubav prema pandama</i>	mogući načini
1	11	$2 \leq n \leq 10$	$1 \leq ljubav \leq 1,000,000$	sva tri načina
2	8	$2 \leq n \leq 1,000$	$1 \leq ljubav \leq 1,000,000$	samo <i>MojiFrendoviSuTiFrendovi</i>
3	8	$2 \leq n \leq 1,000$	$1 \leq ljubav \leq 1,000,000$	samo <i>MiSmoTiFrendovi</i>
4	19	$2 \leq n \leq 1,000$	$1 \leq ljubav \leq 1,000,000$	samo <i>JaSamTiFrend</i>
5	23	$2 \leq n \leq 1,000$	<i>ljubav</i> je uvijek 1	<i>MojiFrendoviSuTiFrendovi</i> i <i>JaSamTiFrend</i>
6	31	$2 \leq n \leq 100,000$	$1 \leq ljubav \leq 10,000$	sva tri načina

Implementacijski detalji

Morate *submitati* točno jednu datoteku, `friend.c`, `friend.cpp` ili `friend.pas`. U ovoj datoteci mora biti implementiran gore opisani potprogram sa dolje navedenim prototipovima. Također, morate *includeati header file* `friend.h` za C/C++.

C/C++ program

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

Paskal program

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

Sample grader

Sample grader prima ulaz sljedećeg oblika:

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

Sample grader ispisat će povratnu vrijednost funkcije `findSample`.