



Ընկեր

Մենք կառուցում ենք սոցիալական ցանց, որը պարունակում է n մարդ, ովքեր համարակալված են $0, \dots, n - 1$ թվերով: Ցանցի մարդկանց որոշ զույգեր կլինեն ընկերներ: Եթե x անձը դառնում է y անձի ընկեր, ապա y անձը նույնպես դառնում է x անձի ընկեր:

Մարդիկ ընդգրկվում են ցանցում n փուլերով, որոնք նույնպես համարակալված են 0 -ից $n - 1$ թվերով: i -րդ անձը ընդգրկվում է i -րդ փուլում: 0 -րդ փուլում 0 -րդ անձն է ընդգրկվում ցանցում, որպես ցանցի միակ անդամ: Հաջորդ $n - 1$ փուլերից յուրաքանչյուրում անձը ընդգրկվում է ցանց *ընդունողի* միջոցով, որը (*ընդունողը*) կարող է լինել արդեն իսկ ցանցում ընդգրկված ցանկացած անձ: i -րդ փուլում ($0 < i < n$) այդ փուլի ընդունողը ընդգրկվող i -րդ անձին կարող է ընդգրկել հետևյալ երեք տարբերակներից մեկով.

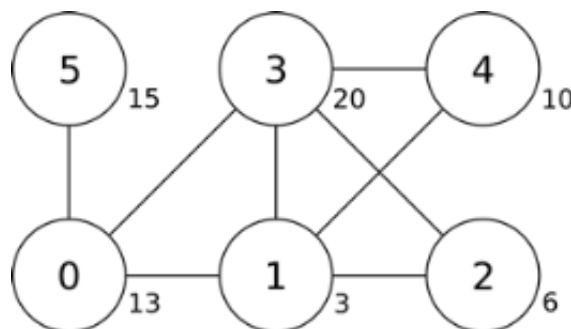
- *IamYourFriend*, որի դեպքում i -րդ անձը դառնում է միայն ընդունողի ընկերը:
- *MyFriendsAreYourFriends*, որի դեպքում i -րդ անձը դառնում է ընդունողի *յուրաքանչյուր* ընկերոջ ընկերը: Այսպես, որ այս տարբերակով ընդգրկվողը *չի դառնում* ընդունողի ընկերը:
- *WeAreYourFriends*, որի դեպքում i -րդ անձը դառնում է ընդունողի ընկերը և ընդունողի *յուրաքանչյուր* ընկերոջ ընկերը:

Ցանցը կառուցելուց հետո մենք ցանկանալու ենք վերցնել տեղեկագրի *նմուշ*, ինչը նշանակում է ընտրել ցանցի մասնակիցներից բաղկացած խումբ: Հաշվի առնելով, որ սովորաբար ընկերները ունենում են նման հետաքրքրություններ, նմուշը չպետք է ընդգրկի միմյանց հետ ընկերներ հանդիսացող որևէ զույգ: Յուրաքանչյուր անձ ունի *վստահության* նկարագիր, որն արտահայտված է որպես դրական ամբողջ և մենք ցանկանում ենք որոշել գումարային մաքսիմալ վստահություն ունեցող նմուշը:

Օրինակ

փուլ	ընդունող	պրոտոկոլ	ընդգրկվելու եղանակը
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)

Ի սկզբանե ցանցը պարունակում է 0 անձին: 1 փուլի ընդունողը (0 անձը) հրավիրում է 1 նոր անձին IamYourFriend տարբերակով և հետևաբար նրանք դառնում են ընկերներ: 2 փուլի ընդունողը (կրկին 0 անձը) հրավիրում է 2 անձին MyFriendsAreYourFriends տարբերակով, ինչը 1 անձին (որը ընդունողի միակ ընկերն է) դարձնում է 2 անձի միակ ընկեր: 3 փուլի ընդունողը (անձ 1) ավելացնում է անձ 3-ին WeAreYourFriends տարբերակով, ինչը անձ 3-ին դարձնում է անձ 1-ի (ընդունողի) ընկեր և 0 և 2 անձերի (ընդունողի ընկերների) ընկեր: 4 և 5 փուլերը նույնպես նշված են վերևի աղյուսակում: Վերջնական ցանցը պատկերված է հաջորդիվ բերված նկարում, որտեղ շրջանագծերի մեջ նշված են մարդկանց համարները, իսկ շրջանագծերին կից թվերն արտահայտում են վստահության նկարագիրը: 3 և 5 անձերին ընդգրկող նմուշի գումարային վստահության նկարագիրը հավասար է $20 + 15 = 35$, ինչը մաքսիմալ հնարավոր գումարային վստահությունն է:



Խնդիր

Տրված է յուրաքանչյուր փուլի նկարագրությունը և յուրաքանչյուր անձի վստահության նկարագիրը: Գտնել մաքսիմալ գումարային վստահությամբ նմուշ: Պահանջվում է իրականացնել `findSample` ֆունկցիան:

- `findSample(n, confidence, host, protocol)`

- n -ը մարդկանց քանակն է:
- confidence-ը n երկարության զանգված է: confidence[i]-ն տալիս է i -րդ անձի վստահության նկարագիրը:
- host-ը n երկարության զանգված է: host[i]-ն i -րդ փուլում ընդունողի համարն է:
- protocol-ը n երկարության զանգված է: protocol[i]-ն ցույց է տալիս, թե i -րդ փուլում ($0 < i < n$) ինչ պրոտոկոլ է օգտագործվել. 0 IamYourFriend-ի դեպքում, 1 MyFriendsAreYourFriends-ի դեպքում, 2 WeAreYourFriends-ի դեպքում:
- Քանի որ 0-րդ փուլում ընդունող չկա, host[0]-ն և protocol[0]-ն սահմանված չեն, Ձեր ծրագրում նրանց մի դիմեք:
- Այս ֆունկցիան պետք է վերադարձնի մաքսիմալ հնարավոր գումարային վստահությունը:

Ենթախնդիրներ

Որոշ ենթախնդիրներ օգտագործում են միայն որոշ պրոտոկոլներ, ինչպես ցույց է տրված հետևյալ աղյուսակում.

Ենթախնդիր	միավոր	n	վստահություն (confidence)	օգտագործվող պրոտոկոլներ
1	11	$2 \leq n \leq 10$	$1 \leq \text{confidence} \leq 1,000,000$	Բոլոր երեք պրոտոկոլները
2	8	$2 \leq n \leq 1,000$	$1 \leq \text{confidence} \leq 1,000,000$	Միայն MyFriendsAreYourFriend
3	8	$2 \leq n \leq 1,000$	$1 \leq \text{confidence} \leq 1,000,000$	Միայն WeAreYourFriends
4	19	$2 \leq n \leq 1,000$	$1 \leq \text{confidence} \leq 1,000,000$	Միայն IamYourFriend
5	23	$2 \leq n \leq 1,000$	All confidence values are 1	MyFriendsAreYourFriend և IamYourFriend
6	31	$2 \leq n \leq 100,000$	$1 \leq \text{confidence} \leq 10,000$	Բոլոր երեք պրոտոկոլները

Իրականացման մանրամասներ

Պետք է հանձնել միայն մեկ ֆայլ, որի անունը պետք է լինի friend.c, friend.cpp կամ friend.pas: Այդ ֆայլում պետք է իրականացնել վերը նկարագրված ֆունկցիան, օգտագործելով հետևյալ նախատիպը: C/C++ ծրագրում պետք է նաև ավելացնել friend.h ֆայլի ընդգրկման հրամանը:

C/C++ ծրագիր

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

Pascal ծրագիր

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

Գրելիղերի օրինակ

Գրելիղերի օրինակը կարդում է մուտքային տվյալները հետևյալ ձևաչափով.

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

Գրելիղերի օրինակը տպելու է findSample ֆունկցիայի վերադարձի արժեքը: