



Φίλος

Χτίζουμε ένα κοινωνικό δίκτυο από n άτομα, αριθμημένα στο διάστημα $0, \dots, n - 1$. Κάποια ζεύγη ατόμων στο δίκτυο θα είναι φίλοι. Αν το άτομο x γίνει φίλος με το άτομο y , τότε το άτομο y γίνεται επίσης φίλος με το άτομο x .

Τα άτομα προστίθενται στο δίκτυο σε n στάδια, τα οποία είναι επίσης αριθμημένα στο διάστημα 0 έως $n - 1$. Το άτομο i προστίθεται στο στάδιο i . Στο στάδιο 0 , το άτομο 0 προστίθεται ως το μοναδικό άτομο του δικτύου. Σε καθένα από τα επόμενα $n - 1$ στάδια, ένα άτομο προστίθεται στο δίκτυο από έναν οικοδεσπότη, ο οποίος μπορεί να είναι οποιοδήποτε άτομο που είναι ήδη στο δίκτυο. Στο στάδιο i ($0 < i < n$), ο οικοδεσπότης για αυτό το στάδιο μπορεί να προσθέσει το εισερχόμενο άτομο i στο δίκτυο χρησιμοποιώντας ένα από τα ακόλουθα τρία πρωτόκολλα:

- *IamYourFriend* κάνει το άτομο i φίλο του οικοδεσπότη μόνο.
- *MyFriendsAreYourFriends* κάνει το άτομο i φίλο με κάθε άτομο που είναι φίλος του οικοδεσπότη εκείνη τη στιγμή. Σημειώστε ότι αυτό το πρωτόκολλο δεν κάνει το άτομο i φίλο του οικοδεσπότη.
- *WeAreYourFriends* κάνει το άτομο i φίλο με τον οικοδεσπότη, και επίσης φίλο με κάθε άτομο που είναι φίλος του οικοδεσπότη εκείνη τη στιγμή.

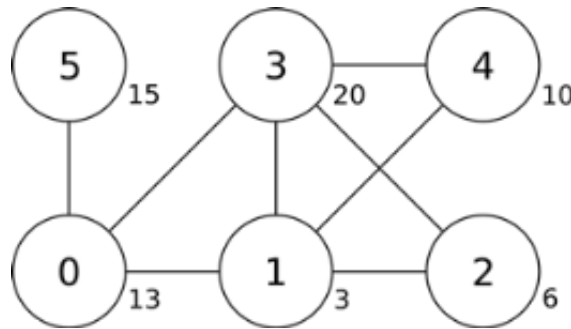
Αφού χτίσουμε το δίκτυο, θα θέλαμε να διαλέξουμε ένα δείγμα για μια μελέτη, δηλαδή, να διαλέξουμε μια ομάδα ατόμων από το δίκτυο. Αφού οι φίλοι έχουν συνήθως παρόμοια ενδιαφέροντα, το δείγμα δεν θα πρέπει να περιλαμβάνει κανένα ζεύγος ατόμων που είναι φίλοι. Κάθε άτομο έχει μία εμπιστοσύνη (confidence), η οποία εκφράζεται ως ένας θετικός ακέραιος, και θα θέλαμε να βρούμε ένα δείγμα με τη μέγιστη συνολική εμπιστοσύνη.

Παράδειγμα

στάδιο	οικοδεσπότης	πρωτόκολλο	σχέσεις φίλων που προστίθενται
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)

Αρχικά το δίκτυο περιέχει μόνο το άτομο 0 . Ο οικοδεσπότης του σταδίου 1 (άτομο 0) προσκαλεί το νέο άτομο 1 μέσω του πρωτοκόλλου *IamYourFriend*, ως εκ τούτου γίνονται φίλοι. Ο οικοδεσπότης του σταδίου 2 (πάλι το άτομο 0) προσκαλεί το άτομο 2 με *MyFriendsAreYourFriends*, το οποίο κάνει το άτομο 1 (το μοναδικό φίλο του οικοδεσπότη) το μοναδικό φίλο του ατόμου 2 . Ο οικοδεσπότης του σταδίου 3 (άτομο 1) προσθέτει το άτομο 3 με *WeAreYourFriends*, το οποίο κάνει το άτομο 3 φίλο του ατόμου 1 (του οικοδεσπότη) και των ατόμων 0 και 2 (φίλοι του οικοδεσπότη). Τα στάδια 4 και 5

παρουσιάζονται επίσης στον πιο πάνω πίνακα. Το τελικό δίκτυο παρουσιάζεται στο ακόλουθο σχήμα, στο οποίο οι αριθμοί μέσα στους κύκλους δείχνουν τις ετικέτες των ατόμων, και οι αριθμοί δίπλα από τους κύκλους δείχνουν την εμπιστοσύνη αυτών των ατόμων. Το δείγμα που αποτελείται από τα άτομα 3 και 5 με συνολική εμπιστοσύνη $20 + 15 = 35$ έχει τη μέγιστη δυνατή συνολική εμπιστοσύνη.



Πρόβλημα

Δοθείσας της περιγραφής κάθε σταδίου και της εμπιστοσύνης κάθε ατόμου, βρείτε ένα δείγμα με τη μέγιστη συνολική εμπιστοσύνη. Πρέπει να υλοποιήσετε μόνο τη συνάρτηση `findSample`.

- `findSample(n, confidence, host, protocol)`
 - `n`: το πλήθος ατόμων
 - `confidence`: πίνακας μεγέθους `n`, όπου `confidence[i]` είναι η εμπιστοσύνη του ατόμου `i`.
 - `host`: πίνακας μεγέθους `n`, όπου `host[i]` είναι ο οικοδεσπότης του σταδίου `i`.
 - `protocol`: πίνακας μεγέθους `n`, όπου `protocol[i]` είναι ο κωδικός πρωτοκόλλου που χρησιμοποιήθηκε στο στάδιο `i` ($0 < i < n$): 0 για `IamYourFriend`, 1 για `MyFriendsAreYourFriends`, και 2 για `WeAreYourFriends`.
 - Αφού δεν υπάρχει οικοδεσπότης στο στάδιο 0, τα `host[0]` και `protocol[0]` δεν είναι καθορισμένα και δεν πρέπει να τα χρησιμοποιήσει το πρόγραμμά σας.
 - Η συνάρτηση πρέπει να επιστρέφει τη μέγιστη δυνατή συνολική εμπιστοσύνη ενός δείγματος.

Υποπροβλήματα

Κάποια υποπροβλήματα χρησιμοποιούν μόνο ένα υποσύνολο των πρωτοκόλλων, όπως φαίνεται στον παρακάτω πίνακα.

υποπρόβλημα	μονάδες	n	εμπιστοσύνη	χρησιμοποιούμενα πρωτόκολλα
1	11	$2 \leq n \leq 10$	$1 \leq \text{confidence} \leq 1,000,000$	Και τα τρία πρωτόκολλα
2	8	$2 \leq n \leq 1,000$	$1 \leq \text{confidence} \leq 1,000,000$	Μόνο <code>MyFriendsAreYourFriends</code>
3	8	$2 \leq n \leq 1,000$	$1 \leq \text{confidence} \leq 1,000,000$	Μόνο <code>WeAreYourFriends</code>
4	19	$2 \leq n \leq 1,000$	$1 \leq \text{confidence} \leq 1,000,000$	Μόνο <code>IamYourFriend</code>

υποπρό-βλημα	μονάδες	n	εμπιστοσύνη	χρησιμοποιούμενα πρωτόκολλα
5	23	$2 \leq n \leq 1,000$	η εμπιστοσύνη είναι πάντα 1	MyFriendsAreYourFriends και IamYourFriend
6	31	$2 \leq n \leq 100,000$	$1 \leq \text{confidence} \leq 10,000$	Και τα τρία πρωτόκολλα

Λεπτομέρειες υλοποίησης

Πρέπει να υποβάλετε ακριβώς ένα αρχείο, με όνομα `friend.c`, `friend.cpp` ή `friend.pas`. Αυτό το αρχείο πρέπει να υλοποιεί το υποπρόγραμμα που περιγράφεται παραπάνω, το οποίο πρέπει να έχει μία από τις παρακάτω επικεφαλίδες. Για τις υλοποιήσεις σε C/C++, το αρχείο σας πρέπει να κάνει `include` το αρχείο επικεφαλίδας `friend.h`.

Πρόγραμμα C/C++

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

Πρόγραμμα Pascal

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

Ενδεικτικός βαθμολογητής

Ο ενδεικτικός βαθμολογητής διαβάζει την είσοδο με την ακόλουθη μορφή:

- γραμμή 1: n
- γραμμή 2: $\text{confidence}[0], \dots, \text{confidence}[n-1]$
- γραμμή 3: $\text{host}[1], \text{protocol}[1], \text{host}[2], \text{protocol}[2], \dots, \text{host}[n-1], \text{protocol}[n-1]$

Ο ενδεικτικός βαθμολογητής θα τυπώσει την τιμή επιστροφής της `findSample`.