

## Sosial Mühəndislik

Məsələ adı	Sosial Mühəndislik
Giriş faylı	İnteraktiv tapşırıq
Çıxış faylı	İnteraktiv tapşırıq
Zaman limiti	5 saniyə
Yaddaş limiti	256 MB

Sosial şəbəkə  $n$  təpədən (vertex) və  $m$  əlaqədən (edge) ibarət istiqamətsiz (undirected) əlaqəli (connected) qrafdır. Burada hər bir təpə bir adamdır və iki təpə arasında əlaqə varsa o təpədəki adamlar dostdurlar. Maria bu sosial şəbəkənin bir üzvüdür. O dostları ilə yarışmağı sevir. Bunun üçün əvvəlcə özü hansısa tapşırığı yerinə yetirir, sonra isə dostlarının birindən bu tapşırığı yerinə yetirməsini istəyir. Daha sonra həmin dost öz dostlarından birinə bu tapşırığı verəcək, və proses bu formada davam edəcək. Hansısa şəxs bu tapşırığı birdən çox dəfə yerinə yetirə bilər, amma istənilən iki dost arasında tapşırıq bir dəfə ötürülməlidir ( $A$  şəxsi  $B$  şəxsinə tapşırıq verərsə, daha sonra nə  $A$   $B$ -yə, nə də  $B$   $A$ -ya tapşırıq verə bilər). Başqa bir sözlə, tapşırıq qrafda heç bir əlaqədən iki dəfə keçmədən gəzəcək. Əgər tapşırıq vermə sırası bir nəfərdədirsə, və həmin adam heç bir dostuna tapşırıq verə bilmirsə o zaman həmin adam udurur. İlk tapşırığı həmişə Maria verir və o çox nadir hallarda udurur. Buna görə də yerdə qalan  $n - 1$  nəfər onu məğlub etmək qərarına gəlirlər. Bu cəhdi idarə etmək öhdəliyi isə sizə verilib.

## İmplementasiya detalları

Aşağıda verilmiş funksiyayı implement etməlisiniz:

```
void SocialEngineering(int n, int m, vector<pair<int,int>> edges);
```

Bu funksiya  $n$  təpə nöqtəsi,  $m$  əlaqəsi olan qrafda oyun oynayacaq. Funksiya qreyder tərəfindən yalnızca bir dəfə çağırılacaq. `edges` çoxluğunda tam olaraq  $m$  sayda  $(u, v)$  cütləri olacaq. Bu,  $u$  və  $v$  təpələri arasında əlaqə var deməkdir. Təpə nöqtələri 1-dən  $n$ -ə qədər nömrələnib. Maria həmişə 1 təpəsindədir. Sizin funksiyanız aşağıdakı metodları çağırabilir:

```
int GetMove();
```

Bu metod Marianın gedişi olduqda çağırılmalıdır (məsələn oyunun başında). Əgər Marianın gedişi

olmadığı halda bu funksiyanı çağırırsanız, o zaman `Wrong answer` cavabı alacaqsınız. Metod aşağıdakı dəyərlərdən birini qaytara bilər:

- $v$  ədədi,  $2 \leq v \leq n$ . Bu o deməkdir ki, Maria tapşırığı  $v$  şəxsinə verdi. Bu, həmişə düzgün gediş olacaq.
- 0, əgər Maria oyundan geri çəkilsə. Əgər mümkün gediş yoxdursa, Maria geri çəkiləcək. Belə olduqda, sizin `SocialEngineering` funksiyanız dayanmalıdır. Bundan sonra `Accepted` cavabı alacaqsınız.

```
void MakeMove(int v);
```

Bu metod Marianın gedişi olmadıqda çağırılmalıdır. Burada, gediş sırası olan şəxs tapşırığı  $v$  təpəsindəki dostuna verir. Əgər bu gediş düzgün deyilsə, və ya Marianın sırası olan vaxt bu metod çağırılsa, o zaman `Wrong answer` cavabı alacaqsınız.

Əgər Marianın udma strategiyası varsa, o zaman `SocialEngineering` funksiyası ilk `GetMove()` çağırışından əvvəl dayanmalıdır. Bu zaman `Accepted` cavabı alacaqsınız.

## Məhdudiyyətlər

- $2 \leq n \leq 2 \cdot 10^5$ .
- $1 \leq m \leq 4 \cdot 10^5$ .
- Qraf əlaqəlidir (connected). Hər bir təpə cütü arasında ən çox bir əlaqə var. Əlaqələr yalnızca müxtəlif təpələr arasındadır.

## Alt tapşırıqlar

Maria həmişə mükəmməl oynayacaq, yəni əgər udma strategiyası varsa həmişə udmaq üçün lazım olan gedişi edəcək. Əgər udma strategiyası yoxdursa sizin səhv gediş etməyiniz üçün müxtəlif yollara əl atacaq. Əgər mümkün gediş yoxdursa, yalnız o zaman geri çəkiləcək (3-cü alt tapşırıq xaric).

1. (15 bal)  $n, m \leq 10$ .
2. (15 bal) Maria xaric hər kəsin ən çox 2 dostu var.
3. (20 bal) Marianın udma strategiyası yoxdursa ilk gedişdə geri çəkiləcək.
4. (25 bal)  $n, m \leq 100$ .
5. (25 bal) Əlavə məhdudiyyət yoxdur.

## Nümunə interaksiya

Sizin gedişiniz	Qreyderin gedişi	İzah
-	<code>SocialEngineering(5, 6, {{1,4},{1,5},{2,4},{2,5},{2,3},{3,5}})</code>	<code>SocialEngineering</code> 5 təpəli və 6 əlaqəli qraf ilə çağırılıb.
<code>GetMove()</code>	4 qaytarır	Maria 4 nömrəli şəxsə tapşırıq verir.
<code>MakeMove(2)</code>	-	4 nömrəli şəxs tapşırığı 2 nömrəli şəxsə verir.
<code>MakeMove(5)</code>	-	2 nömrəli şəxs tapşırığı 5 nömrəli şəxsə verir.
<code>MakeMove(1)</code>	-	5 nömrəli şəxs tapşırığı Mariaya verir.
<code>GetMove()</code>	0 qaytarır	Marianın mümkün gedişi yoxdur, buna görə də geri çəkilir.
<code>Return</code>	-	Siz oyunda qalib gəldiniz və <code>SocialEngineering</code> funksiyasını dayandırmalıdır.

Sizin gedişiniz	Qreyderin gedişi	İzah
-	<code>SocialEngineering(2, 1, {{1,2}})</code>	<code>SocialEngineering</code> 2 təpəli və 1 əlaqəli qraf ilə çağırılıb.
<code>Return</code>	-	Marianın bu qrafda udma strategiyası var, ona görə də geri çəkilmək üçün heç bir <code>GetMove()</code> çağırışı etmədən funksiyanı dayandırmalısınız.

## Nümunə Qreyder

`SocialEngineering.zip` içində verilmiş, `grader.cpp` giriş verilənlərini standart girişdən aşağıdakı formada oxuyur:

- Birinci sətirdə qrafda təpələrin sayı olan  $n$ , və əlaqələrin sayı olan  $m$  ədədi var.
- Növbəti  $m$  sətirin hər birində  $u$  və  $v$  tam ədədləri var. Bu o deməkdir ki,  $u$  və  $v$  təpələri arasında əlaqə var.

Nümunə qreyder giriş verilənlərini oxuyur və iştirakçının həllindəki `SocialEngineering` funksiyasını çağırır. Qeyd edək ki, nümunə qreyderdə Marianın udma strategiyası yoxdur və

yalnızca nümunə interaksiya üçün verilib.

Nümunə qreyderi öz həllinizlə yoxlamaq üçün aşağıdakı əmri terminalda istifadə edə bilərsiniz:

```
g++ -std=gnu++11 -O2 -o solution grader.cpp solution.cpp
```

Burada `solution.cpp` sizin CMS-ə göndərəcəyiniz fayldır. Qoşmadakı giriş faylı ilə bir yerdə proqramı işlətmək üçün aşağıdakı əmri terminalda yazın:

```
./solution < input.txt
```