Többségi képviselő kiválasztása

Iskolád tanulói két csoportba tartoznak. Tudjuk, hogy az egyik csoportban többen vannak, mint a másikban, ezt nevezzük többségi csoportnak. Ki kell választani egy tanulót, aki a többségi csoporthoz tartozik. Ehhez egyetlen műveletet használhatunk, nevezetesen két tanulótól megkérdezhetjük, hogy ugyanabba a csoportba tartoznak-e.

Feladat

Olyan programot kell írni, amelyik a lehető legkevesebb kérdéssel meghatároz egy többségi csoporthoz tartozó tanulót. A tanulókat sorszámukkal azonosítjuk.
A megoldáshoz a `query` modul három művelete használható.

Könyvtári műveletek

Size A tanulók n számát adja. Ezt kell először hívni.

Member Két tanuló sorszámát kell argumentumként megadni, és a függvény 1 értéket ad, ha a két tanuló ugyanazon csoport eleme, egyébként 0-át.

Answer Ezzel a művelettel kell közölni a kiválasztott, többségi csoportba tartozó tanuló sorszámát. Végrehajtásával a program végrehajtása befejeződik.

A query modul műveletei Pascal nyelv esetén

- `function Size:integer;`
- `function Member(x, y: integer): integer;`
- `procedure Answer(x: integer);`

A query modul műveletei C/C++ nyelv esetén

- `int Size();`
- `int Member(int x, int y);`
- `void Answer(int x);`

Feltételek és korlátozások

- A tanulók n számára $5 \leq n \leq 30000$ teljesül és n páratlan szám.
- Programod nem írhat és nem olvashat egyetlen fájlt sem, beleértve a standard bemenetet és kimenetet!
- A megoldást csak akkor fogadják el, ha a tanulók bármely olyan diszjunkt A és B részhalmazára, amely kompatibilis az általad feltett kérdésekkel, a közölt megoldás a nagyobb elemszámú részhalmazban van. A válaszadó arra kényszerít, hogy szükséges számú kérdést tegyél fel.

Gyakorlás

A könyvtári modul úgy használható, hogy a standard bemenet első és egyetlen sorába a tanulók n számát kell írni, ami páratlan szám kell legyen! A program a standard kimenetre kiírja a végrehajtott kérdéseket a válasszal, továbbá a választ és annak helyes vagy hibás voltát. Hibás válasz esetén azt is kiírja, hogy miért hibás a válasz.

Példa

Ha a bemenet 7, akkor a kimenet az alábbi lesz:

```
Size=7
Member(1,2)=1
Member(3,4)=1
Member(2,4)=0
Member(5,6)=1
Valaszod=6, Helyes
Többségi csoport:
3..6
Kissebségi csoport:
1 2 7
A végrehajtott kérdések száma: 4
A lehetséges maximális pontszám: 3
Pontszámod: 3
```

Azonban az 1 válasz nem elfogadható, mert minden feltett kérdésre a többségi csoport a $\{2,5,6,7\}$, a kisebbségi pedig a $\{1,3,4\}$ halmaz, akkor a `Member` függvény ugyanazt eredményezné, de 1 nem eleme a $\{2,5,6,7\}$ többségi csoportnak.

Pontozás

Helyes válasz esetén a kapott pontszám: $\max(0, n-k)$, ha a programod k `Member` műveletet hajtott végre.

Megoldás

Jelölje $H = \{1, \ldots, n\}$ a tanulók halmazát. Azt mondjuk, hogy egy $A \subseteq H$ részhalmaz **homogén** részhalmaz,

ha A minden eleme ugyanabba a csoportba tartozik, azaz ha $(\forall x, y \in A)(Member(x,y) = 1)$. Azt mondjuk, hogy $U, V \subseteq H$ **ellentétes** részhalmazok, ha egyrészt U és V homogén, továbbá U minden eleme az egyik, V minden eleme a másik csoportba tartozik, azaz ha $(\forall x \in U)(\forall y \in V)(Member(x,y) = 0)$.

Kérdésekből származó ismeret ábrázolása

1. Észrevétel

Megmutatjuk, hogy a feltett kérdésekből származó ismeret ábrázolható diszjunk ellentétes részhalmaz-párok halmazaként. Tehát

$$I = \{(U_1, V_1), \ldots, (U_k, V_k)\}$$

alakban, ahol U_i, V_i ellentétes, továbbá

$$\bigcup_{i=1}^{k} U_i \cup \bigcup_{i=1}^{k} V_i = \{1, \ldots, n\}$$

Bizonyítás. Kezdetben nincs semmi ismeretünk, tehát $(\{i\}, \emptyset)$ párok $(i = 1, \ldots n)$ alkotják az ismeretet.
Tegyük fel, hogy az eddig végrehajtott `Member` műveletek által szerzett ismeret megadható az

$$I = \{(U_1, V_1), \ldots (U_k, V_k)\}$$

halmazzal és `Member(x,y)` kérdést tettük fel. x is és y is pontosan az egyik halmazba esik az U_i és V_i halmazok közül. Tehát az alábbi négy eset lehetséges:

$$x \in U_i, \text{ és } y \in U_j$$

$$x \in U_i, \text{ és } y \in V_j$$

$$x \in V_i, \text{ és } y \in U_j$$

$$x \in V_i, \text{ és } y \in V_j$$

Az általánosság megszorítása néélkül feltehetjük, hogy az 1. esetről van szó. Ha `Member(x,y)=1` akkor vegyük az

$$U = U_i \cup U_j \text{ és } V = V_i \cup V_j$$

halmazokat, ha pedig `Member(x,y)=0`, akkor a

$$U = U_i \cup V_j \text{ és } V = V_i \cup U_j$$

halmazokat. Ekkor az új ismeretet úgy ábrázolhatjuk, hogy az I halmazból törüljük az (U_i, V_i) és (U_j, V_j) párokat és bevesszük az (U, V) párt.

2. Észrevétel

Ha $U, V \subseteq H$ homogén ellentétes részhalmazok, továbbá U és V elemszáma megegyezik ($|U| = |V|$), akkor H-ból törölve az U és V elemeit ($H := H - U - V$), a megmaradt halmaz továbbra is tartalmaz egy többségi csoporthoz tartozó elemet.

3. Észrevétel

Ha mindig olyan `x,y` párra hajtjuk végre at `Member(x,y)` műveletet, amelyek teljesül, hogy az őket tartalmazó homogén részhalmazok elemszáma megegyezik, akkor az ismeret árázolásában minden V_i halmaz üres halmaz lesz:

$$I = \{(U_1, \emptyset), \ldots, (U_k, \emptyset)\}$$

Kezdetben $U_i = \{i\}$ és $V_i = \emptyset$. Ha `Member(x,y)` műveletet előtt teljesült a feltétel, akkor utána is teljesül. Valóban, ha `Member(x,y)=1`, akkor $U = U_i \cup U_j$ lesz, ha pedig `Member(x,y)=0`, akkor az 1. éstrevétel alapján töröljük I-ből $(U_i, \emptyset)$ és $(U_j, \emptyset)$ párt, és töröljük ki a H alaphalmazból is U_i és U_j elemeit.
Tehát ekkor minden U_i homogén halmaz elemszáma 2-hatvány, továbbá, ha bármely két U_i és U_j halmaz elemszáma megegyezik, akkor mindkettő egyelemű.

Mikor van elegendő ismeretünk a válasz megadására?

Ha az

$$I = \{(U_1, V_1), \ldots, (U_k, V_k)\}$$

ismerettel rendelkezünk, és teljesül, hogy

$$|U_j| + \sum_{i=1}^{k} \min(|U_i|, |V_i|) - |V_j| > n/2$$

ahol U_j a legnagyobb elemszámú részhalmaz, akkor elegendő ismerettel rendelkezünk, meg tudunk adni egy többségi elemet.
Ha mindig azonos elemszámú homogén részhalmazokban lévő elemkre kérdezünk rá, akkor az

$$I = (U_1, \emptyset), \ldots, (U_k, \emptyset)$$

ismeret elegendő, ha

$$|U_1| \geq \sum_{i=2}^{k} |U_i|$$

feltéve, hogy

$$|U_i| \geq |U_{i+1}| i = 1, \ldots k - 1$$

Ekkor U_1 bármelyik eleme többségi elem lesz. Ekkor, mivel a minden V_i részhalmaz üres, ezért az I ismeretben elég csak az U_i részhalmazokat megadni.

Elvi algoritmus

$I := \emptyset$
x:=0
ciklus amíg I nem biztos ismeret
  x:=x+1
  y:=x+1
  ha $Member(x, y) = 1$ **akkor**
    $U := \{x, y\}$;
    ciklus amíg van olyan $U_i \in I$, hogy $|U| = |U_i|$
      $y := U_i$ tetsőleges eleme;
      $I := I - \{U_i\}$
      ha Member(x,y) **akkor**
        $U := U \cup U_i$
      egyébként
        kilépés a ciklusból
      elágazás vége
    ciklus vége
  elágazás vége
ciklus vége

Belátható, hogy az algoritmus legfeljebb $n - b(n)$ kérdéssel megtalál egy többségi elemet, ahol $b(n)$ az n szám kettes számrendszerbeli leírásában az 1-es jegyek száma. A válaszadó tudja kényszeríteni ennyi kérdésre a kérdezőt. Így működik a mintamegoldáshoz adott `query` modul, tehát a legtöbb kérdésre kényszerít.
A megvalósítás során elég minden részhalmazt egy elemmel reprezentálni. Mivel adott k-ra legfeljebb egy 2^k elemszámú részhalmaz van az I ismerethalmazban, így $B[]$ bitvektorral megadható, és ezért egyszerűen eldonthető, hogy adott k-ra van-e 2^k elemszámú részhalmaz I-ben.

Megvalósítás C++ nyelven

```cpp
#include "query.h"
#define MaxN        30000    //max. méret
#define MaxK        20       //MaxN<=2^MaxK
int main(){
  int N;               //a tanulók száma
  int M;               //az aktuális elemszám
  int Fel;             //az aktuális elemszám fele
  bool B[MaxK];        //B[k]=true, akkor és csak akkor, ha van 2^k elemszámú részhalmaz
  int Rep[MaxK];       //Rep[k] a 2^k elemszámú részhalmaz egy eleme
  int Pow2[MaxK];      //Pow2[k]=2^k 2-hatványok
  int L;               //a legnagyobb elemszámú részhalmaz elemszáma 2^L
  int i, k;
  Pow2[0] = 1;
  for (k = 1; k <= MaxK; k++){
    Pow2[k] = Pow2[k-1] << 1;
    B[k]=false;
  }
  N = Size();
  M = N - 1;
  Fel = M/2 + 1;
  L = 0;
  i = 0;
  while (i < N){
    k = 0;
    B[0] = true;
    Rep[0] = ++i;
    i++;   //
    if (i > N) break;
    while (B[k]){                  //van két 2^k elemszámú részhalmaz
      if (Member(Rep[k],i)==1){   //egyesítsük a két 2^k elemszámú részhalmazt
         B[k] = false;
         k++;
         if (k>L) L=k;            //új legnagyobb elemszámú részhalmaz
         continue;
      }
      M -= Pow2[k+1];             //M:=M-2^(k+1)
      Fel -= Pow2[k];             //Fel:=Fel-2^k
      B[k] = false;               //töröljük a részhalmazt
      if (k==L)                   //L aktualizálása
         while (L>0 && !B[L]) L--;
      k = -1;
      break;
    }//while
    if (k>=0) {
      B[k] = true;                //form a new subgroup having 2^k elements
      Rep[k] = i;                 //i az új részhalmaz reprezentálása
    }
    if (L>0 && Pow2[L]>=Fel)      //van elég ismeretünk
      break;
  }
  Answer(Rep[L]);                 //a legnagyobb részhalmaz egy eleme a megoldás
}//main
```