­­A Bran kastélyba látogató turistákat elfogtat Drakula gróf. A turisták között van egy varázsló, aki a következő megállapodásra jutott a gróffal: egy különleges trükköt mutat be, és ha sikerül, akkor Drakula mindenkit elenged.

A trükk elvégzéséhez a varázslónak két segédre van szüksége. A trükk elkezdése után nem beszélhet a segédeivel, és a segédek sem egymással. A varázsló 2N + 1 kártyából álló kártyapaklit ad a grófnak, mindegyik kártyára 0 és 2N közötti különböző szám van írva. A gróf kivesz egy kártyát és elrejti. Ezután a maradék 2N kártya közül a gróf kiválaszt N darab kártyát és az első segédnek adja, és a maradék N darabot a másodiknak. Ezt követően mindkét segéd kiválaszt két kártyát a sajátjaiból és megmutatja ezeket a varázslónak (de a másik segéd nem látja). A 4 kártya ismeretében a varázsló megsejti, hogy milyen kártya van Drakulánál!

Segíts a varázslónak és ne engedd, hogy a turisták a vámpír vacsorája legyenek!

A programod minden tesztre háromszor fut le. Az első futás alkalmával az első segéd szerepét játssza. A második futáskor a második segéd szerepét játssza. A harmadik futásnál pedig a varázslóét.

**Bemenet**

A trick.in szöveges bemeneti állomány első sora a tesztesetek T számát tartalmazza (ahányszor a trükköt végrehajtják). A második sor egy R egész számot tartalmaz, amelynek értéke 1, 2, 3 lehet, jelezvén az aktuális szereplőt minden tesztesetre.

Ha R=1, akkor a program az első segéd szerepében fut; ha R=2, akkor a másodikéban. Mindkét esetben a 2i+1. sor (1≤i≤T) egy Ni egészet tartalmaz, ami azt jelenti, hogy az i. tesztesetben a kártyák száma 2Ni+1. A 2i+2. sor (1≤i≤T) Ni egész számot tartalmaz, az aktuális segéd által kapott kártyák értékét az i. tesztesetben.

Ha R=3, akkor a program a varázsló szerepét játssza. A 2i+1. sor (1≤i≤T) egy Ni egész számot tartalmaz ugyanazzal a jelentéssel, mint a segédeknél volt. A 2i+2. sor (1≤i≤T) négy egész számot tartalmaz, az első kettő az a két szám, amit a programod az R=1 esetén írt ki, a második kettő pedig az, amit a programod az R=2 esetén írt ki az i. esetben. A kártyákat ugyanolyan sorrendben kapod meg, ahogy kiírtad a megfelelő futásnál.

**Kimenet**

Ha R=1 vagy R=2, akkor a programodnak a trick.out szöveges kimeneti állomány i. sorába (1≤i≤T) két, szóközzel elválasztott egész számot kell írnia, annak a két kártyának a számát, amit a segéd mutat a varázslónak az i. tesztesetben. A két szám különböző legyen, és azok közül való, amelyeket a segéd a bemeneti fájlban kapott.

Ha R=3, akkor a programod az i. sorba (1≤i≤T) azt a számot írja, amelyet Drakula elrejtett az i. esetben.

**Korlátok**

* 6 ≤ Ni ≤ 1 234 567 (1 ≤ i ≤ T)
* SN ≤ 1 234 567, ahol SN = N1 + N2 + ... + NT
* 29 pontot érő tesztek esetén, Ni = 6 (1 ≤ i ≤ T)
* További 19 pontot érő tesztek esetén, 6 ≤ Ni ≤ 30 (1 ≤ i ≤ T) és SN ≤ 123 456
* További 30 pontot érő tesztek esetén, 6 ≤ Ni ≤ 500 (1 ≤ i ≤ T), SN ≤ 123 456 és legfeljebb 10 teszteset lesz, ahol Ni > 50

**Példa**

|  |  |  |
| --- | --- | --- |
| trick.in | trick.out | Megjegyzés |
| 2  1  6  6 1 2 5 7 10  6  9 8 2 0 4 6 | 1 2  8 4 | A három táblázatsor a három futtatásnak felel meg.  A trükköt kétszer adják elő. Az első esetben, az első segéd az 1, 2, 5, 6, 7, 10 kártyákat kapja, és a varázslónak az 1 és 2 kártyát mutatja ebben a sorrendben. A második segéd a 0, 3, 4, 8, 9, 12 kártyákat kapja, és a varázslónak a 4 and 3 kártyákat mutatja ebben a sorrendben. A varázsló ezek alapján kitalálja, hogy Drakukánál a 11-es kártya van. |
| 2  2  6  3 0 4 9 12 8  6  7 1 11 10 3 5 | 4 3  1 3 |
| 2  3  6  1 2 4 3  6  8 4 1 3 | 11  12 |