

1. feladat: Magasugró verseny (15 pont)

Egy magasugró versenyen a versenyzők különböző magasságokat próbálnak átugorni, legalább egyet minden versenyző ugrik. Nem kell mindenkinek minden magasságot megpróbálni, de egy magasságon maximum háromszor próbálkozhat. Ha egyik sem sikerült, akkor kiesik a további versenyből. A verseny végeredményeként az A versenyző előrébb végez a B versenyzőnél, ha

- nagyobb magasságon van sikeres ugrása, vagy
- azonos magasságon kevesebb kísérletből van sikeres ugrása, vagy
- visszafelé haladva a korábbi kísérleteken az előző két eset valamelyike fennáll;
- ha a versenyzők minden magasságon azonos eredményt értek el, akkor holtverseny van közöttük.

Készíts programot, amely a verseny ugrásai alapján megadja a versenyzők végső sorrendjét!

*Bemenő adatok:*

A MAGAS.BE állomány első sorában a versenyzők száma ( $2 \leq N \leq 40$ ) van, a további soraiban pedig soronként egy-egy magasságon a versenyzők eredményei (legfeljebb 20 sor). Minden sor elején az ugrott magasság, majd a sor végéig az egyes versenyzők kísérletei: egy szóköz, a versenyző sorszáma, a J vagy az R betű (a jó vagy a rossz ugrások azonosítására).

Példa:

```
5
215 1J 3J 4J 5J
220 1R 2R 4R 5R 1R 2R 4R 5R 1J 2R 4R 5R
225 1J 3J
230 1R 3R 1R 3R 1R 3R
```

*Kimenő adatok:*

A MAGAS.KI állományba legfeljebb N sort kell írni, az I. sorba az I. helyen végzett versenyző sorszámát. Ha az I. helyen holtverseny van, akkor az I. sorban több versenyző sorszám szerepel, egy-egy szóközzel elválasztva.

Példa:

```
1
3
4 5
2
```

2. feladat: Kapcsoló (18 pont)

Három különböző nyomtatót szeretnénk számítógépünkhöz kapcsolni, de a gépnek csak két portja van. Sikerült vásárolni egy olyan 2→3 kapcsolót, amelynek két bemenete a két porthoz, három kimenete pedig a három nyomtatóhoz köthető. Így mind a három nyomtatóra nyomtathatunk, csak a nyomtatás előtt egy kapcsológombbal a megfelelő port→nyomtató kapcsolatot ki kell alakítani, ha a kívánt nyomtató éppen nincs egyik porthoz sem kötve a 2→3 kapcsoló által.

Adott nyomtatandó anyagoknak egy sorozata, megjelölve, hogy az anyagot melyik nyomtatón kell kinyomtatni. A nyomtatókat az A, B és C karakterekkel azonosítjuk. A nyomtatást a megadott sorrendben kell elvégezni, és egyszerre csak egy anyagot nyomtathatunk. Kezdetben egyik nyomtató sincs porthoz kapcsolva.

Készíts programot, amely kiszámítja, hogy minimálisan hány kapcsolás szükséges a bemenetként megadott nyomtatási igény végrehajtásához!

*Bemenő adatok:*

A KAPCSOLO.BE szöveges állomány első sorában a nyomtatandó anyagok száma ( $1 \leq N \leq 1000$ ) van. A következő sorban N karakterrel adjuk meg az N anyag nyomtató-azonosítóját (A, B és C betűk, elválasztójel nincs közöttük).

*Kimenő adatok:*

A KAPCSOLO.KI szöveges állomány a nyomtatáshoz szükséges kapcsolások minimális számát tartalmazza (egyetlen sor, a szám előtt lehetnek bevezető szóközök).

Példa:

Bemenet	Kimenet
10	4
<b>AABACBAAB</b>	

(A példában vastagon szedtük a kapcsolások lehetséges helyét.)

3. feladat: Dominók (16 pont)

Minden egyes dominón két szám található. A dominójátékban úgy tehetünk egymás mellé dominókat, hogy az egyikén levő jobboldali szám illeszkedjen a mellette levő dominó baloldali számához. Például:

1	5	5	3
---	---	---	---

Ehhez a párhoz jobbról az alábbi dominó úgy illeszthető, ha megfordítjuk:

Tegyük le a játék szabályainak megfelelően az összes használható dominót úgy, hogy az első letevése után csak az egyik irányba (tőle jobbra) tehetünk!

8	3
---	---

*Bemenő adatok:*

A DOMINO.BE állomány első sorában a felhasználható dominók száma van ( $1 \leq N \leq 100$ ). A következő N sorban pedig egy-egy dominóleírás, a rajta levő két szám, egy szóközzel elválasztva.

*Kimenő adatok:*

Ha a dominókat le lehet rakni a szabályoknak megfelelően, akkor a DOMINO.KI állomány szerkezete azonos a DOMINO.BE állománnyal, de a dominók sorrendje a lerakás sorrendjével azonos. A dominókon levő két szám sorrendje is a lerakás sorrendje legyen! Ha a dominókat nem lehet szabályosan lerakni, akkor a DOMINO.KI állomány egyetlen sorból álljon, melyben egy 0 van.

Példa: DOMINO.BE

4  
1 5  
3 2  
3 4  
2 1

DOMINO.KI

4  
5 1  
1 2  
2 3  
3 4