

1. feladat: Futár (40 pont)

Egy vállalat futárokat alkalmaz a bizalmas levelek továbbítására. Több városban is lehet futár, és ha egy adott cél városba kell levelet vinni, akkor mindig az a futár viszi a levelet, amelyik a leghamarabb célhoz ér. Minden futár egy nap alatt csak olyan úton haladhat, amely legfeljebb K várost érint.

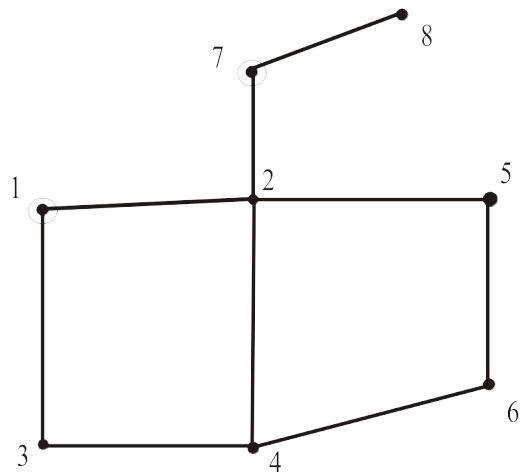
Készíts programot (FUTAR.PAS, FUTAR.C vagy FUTAR.CPP), amely megadja azon városok sorszámát, amelyekbe 1 nap alatt nem juthat el futár, valamint azt, hogy hány nap kell, hogy biztosan eljusson mindenhova!

A FUTAR.BE állomány első sorában a városok száma ($1 \leq N \leq 200$), a futár által egy nap alatt bejárható városok száma ($1 \leq K \leq N$) és a városok közötti utak száma ($1 \leq M \leq 10000$) van. A következő M sor mindegyike 2 egész számot tartalmaz, amelyek egy-egy utat írnak le: milyen sorszámú városból ($1 \leq X \leq N$) milyen sorszámú városba ($1 \leq Y \leq N$) vezet. Az állomány utolsó sorában azon városok sorszámát szerepel egy-egy szóközzel elválasztva, ahonnan indulhat a futár. Az utak kétirányúak, nincs olyan város, ahova futár nem juthat el.

A FUTAR.KI állomány első sorába azon városok sorszámát kell írni, egy-egy szóközzel elválasztva, ahova nem juthat el futár 1 nap alatt. A második sorba azon napok legkisebb számát kell írni, ahány nap alatt a legtávolabbi városba is elérhet futár

Példa:

FUTAR.BE	FUTAR.KI
8 2 9	6
1 2	2
1 3	
2 4	
3 4	
2 5	
4 6	
5 6	
2 7	
7 8	
1 7	



2. feladat: Kannák (30 pont)

Egy gazdának három különböző űrtartalmú tejeskannája van, amelyekbe teli állapotban A, B és C liter tej fér. Van továbbá egy negyedik kannája, ennek az űrtartalmát nem ismeri, csak azt tudja, hogy ez a legnagyobb kannája. A gazda észrevette, hogy a kannák közötti öntögetésekkel bizonyos mennyiségű tejet ki tud mérni. Az öntögetések során csak azt kell betartania, hogy tudja, hogy mennyi tej marad abban a kannában, amelyikből tölt és mennyi lesz abban, amelyikbe tölt. Szeretné megtudni, hogy melyek azok a mennyiségek, amelyeket nem tud így kimérni.

Írj programot (KANNAK.PAS, KANNAK.C vagy KANNAK.CPP), amely kiszámítja, hogy öntögetéssel milyen mennyiségű tejet nem tud a gazda elkülöníteni a negyedik kannában. Kezdetben a legnagyobb, ismert űrtartalmú kanna tele van, a többi pedig üres.

A KANNAK.BE állomány egyetlen sorában három pozitív egész szám van, a három kanna űrtartalma. A számok értéke nem nagyobb, mint 39.

A KANNAK.KI állomány egyetlen sorába azokat az értékeket kell kiírni, amelyek a kezdetben meglévő tej mennyiségénél kisebbek, de nem állíthatók elő kannák közötti öntögetéssel. Ha minden lehetséges érték előállítható, akkor az egyetlen 0 számot kell kiírni.

Példa:

KANNAK.BE

13 11 5

KANNAK.KI

4 9

3. feladat: Üveg (30 pont)

Üvegek újrahasznosítását végző üzembe ládákban érkeznek az üvegek. A feldolgozás végett külön kell válogatni a fehér, a zöld és a barna üvegeket. Minden ládáról tudjuk, hogy hány fehér, zöld illetve barna üveget tartalmaz. A válogatást úgy végzik el, hogy mindhárom fajta számára kijelölnek egy-egy ládát, és minden más ládából az adott fajtájú üveget ebbe a kijelöltbe rakják át.

Készíts programot (UVEG.PAS, UVEG.C vagy UVEG.CPP), amely megad három különböző ládát a fehér, a zöld és a barna üvegek számára úgy, hogy az átpakolás során a lehető legkevesebb üveget kelljen mozgatni!

Az UVEG.BE állomány első sorában egyetlen egész szám van, a ládák N ($3 \leq N \leq 20000$) száma. A következő N sor mindegyike három egész számot tartalmaz; $F Z B$, ($0 \leq F, Z, B \leq 3000$), amelyek egy ládában lévő fehér, zöld, illetve barna üvegek száma.

A UVEG.KI állomány első és egyetlen sorába három ládasorszámot kell írni (egy-egy szóközzel elválasztva), amelyek .rende a fehér, a zöld illetve a barna üvegek számára kijelölt ládák. A ládákat a bementi állománybeli sorszámukkal azonosítjuk, azaz az i -edik láda tartalma az állomány $i+1$ -edik sorában van megadva.

Példa:

UVEG.BE	UVEG.KI
5	4 1 2
1 6 2	
2 4 3	
1 3 0	
9 2 1	
1 2 1	