

**A. 782.** Bizonyítsuk be, hogy egy síkbarajzolható egyszerű gráf éleit mindig lehet úgy irányítani, hogy minden pont kifoka legfeljebb három legyen.

*Angol versenyfeladat*

**Beküldési határidő: 2020. október 10.**

**Elektronikus munkafüzet:** <https://www.komal.hu/munkafuzet>



## Informatikából kitűzött feladatok



**I. 514.** A helyiértékes számrendszerekben a számok számjegyeit a számrendszer alapszámának megfelelő hatványával szorozzuk, hogy megkapjuk a szám értékét. Például a 143 esetében  $1 \cdot 10^2 + 4 \cdot 10^1 + 3 \cdot 10^0$ , azaz  $100 + 40 + 3$  a szám értéke. A negatív számokat egy előjellel jelöljük, de a felírás itt is ugyanúgy történik. Ha azonban alapszámnak egy negatív számot választunk, akkor nem lesz szükségünk előjelre. Legyen a számrendszer alapszáma  $-10$ . Ekkor a  $-10$  alapú számrendszerben felírt szám számjegyeit  $-10$  hatványaival szorozzuk, tehát a  $345_{-10}$  szám értéke  $3 \cdot (-10)^2 + 4 \cdot (-10)^1 + 5 \cdot (-10)^0$ , vagyis 265. Könnyen belátható, hogy a  $-10$  alapú számrendszerben is egyértelmű a számok felírása, de nincs szükség a negatív számok esetében az előjelre. Például  $-25 = -30 + 5$ , tehát  $-10$  alapú számrendszerben  $35_{-10}$ .

Készítsünk programot, amely  $N$  darab 10-es számrendszerben megadott számot átvált  $-10$ -es számrendszerbe. A program a standard bemenet első sorából olvassa be az átváltandó számok darabszámát ( $1 \leq N \leq 100$ ), majd a következő  $N$  sorból az átváltandó  $A$  számokat ( $|A| \leq 10^9$ ), és írja a standard kimenet  $N$  darab sorába a számok felírását  $-10$  alapú számrendszerben.

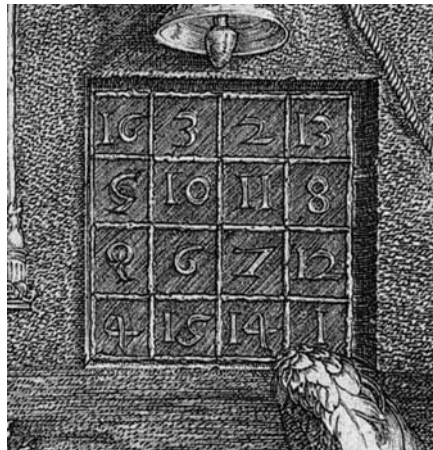
*Példa:*

Standard bemenet	Standard kimenet
3	1832
-228	166
46	361
241	

Beküldendő egy tömörített `i514.zip` állományban a program forráskódja és rövid dokumentációja, amely megadja, hogy a forrásállomány melyik fejlesztői környezetben fordítható.

**I. 515.** A bűvös négyzetek  $N \times N$  egész számot tartalmaznak négyzetes elrendezésben. Minden szám egyedi, nem ismétlődik, és bármely sorban, oszlopban, valamint az átlókban szereplő számok összege ugyanaz az érték.

A bűvös négyzetek régóta ismertek. Egy híres bűvös négyzetet látunk Albrecht Dürer (1471–1528) magyar származású német festő és grafikus *Melankólia* című metszetének jobb felső sarkában. A négyzet alsó sorában középen szereplő két érték összeolvasva 1514, amely a kép keletkezésének éve.



Kaptunk egy  $4 \times 4$ -es bűvös négyzetet, amely elképzelhető, hogy hibás. Ismerjük a minta készítőjét, aki néha felcserél oszlopokat, alkalmanként felcserél sorokat, de előfordul, hogy egy számot eltéveszt.

A számokat rögzítettük a `durer.txt` tabulátorral tagolt, UTF 8 kódolású állományban. Értékeljük ki a bűvös négyzetet táblázatkezelő segítségével az alábbiak szerint:

- Adjuk meg, hogy a bűvös négyzet helyesen van-e kitöltve.
- Ha számtévesztés történt, akkor adjuk meg a cellahivatkozást, ahol a hibás szám van és írjuk ki mellé a helyes értéket. A hibás cella háttérét feltételes formázással színezzük ki.
- Ha a számok helyesek soronként és oszloponként, de fel vannak cserélve, akkor írjuk ki, hogy az átlók tévesek.

*Minta:*

	A	B	C	D	G	H
1	16	3	2	13		Téves szám!
2	5	10	11	8		Hiba a C3 cellában van.
3	9	6	8	12		Helyes érték a 7 lenne.
4	4	15	14	1		
7						

Beküldendő egy `i505.zip` tömörített állományban a munkafüzet, valamint egy rövid dokumentáció, amelyből kiderül az alkalmazott táblázatkezelő neve és verziószáma.

**I. 516.** Mari néni és Bözsi néni is fogyni szeretne. A nagyobb motiváció érdekében figyelemmel kísérik egymás testsúlyának alakulását. Az adatokat minden hétfőn rögzítik a `fogyi.txt` állományba. A fájl sorai rendre Mari néni majd Bözsi néni tömegét tartalmazzák kg-ban egészen kerekítve az egymást követő heteken egy szóközzel elválasztva. A fájl legfeljebb egy év (53 hét) adatait tartalmazza.

87 85

86 88

A példában a második héten Mari néni 86 kg, Bözsi néni pedig 88 kg volt. Készítsünk `fogyokura` néven programot az alábbiak szerint. Minden esetben írassuk ki a feladat számát is.

1. Olvassuk be és tároljuk el az adatokat.
2. Kérjük be egy hét sorszámát, majd írassuk ki, hogy ezen a hétfőn hány kilogramm volt a két hölgy.
3. Írjuk ki a képernyőre, hogy hány olyan hét volt, amikor Mari néni tömege meghaladta Bözsi néniét.
4. Számítsuk ki, hogy mennyi volt a mérés ideje alatt Bözsi néni átlagos tömege. Az eredményt két tizedesjegy pontossággal írassuk ki a képernyőre.
5. Vizsgáljuk meg, hogy melyik héten volt Mari néni tömege a legnagyobb. Ha több ilyen hét is volt, akkor mindegyiket írassuk ki a képernyőre.
6. Keressük meg, hogy mikor csökkent az előző héthez képest az egyik, illetve a másik hölgy tömege. Az eredményt egy-egy szóközzel elválasztva a következő formában írassuk ki a `csokken.txt` nevű szöveges állományba:

Mari néni: 12-13 13-14 15-16

Bözsi néni: 2-3 10-11 29-30

Például Mari néni súlya a 12. hétről a 13. hétre csökkent, majd a 13. hétről a 14. hétre is stb.

7. Írjuk egy táblázatban a képernyőre, hogy adott tömegeket mely heteken mért Mari néni. Például:

86 kg: 2 4 8 12

87 kg: 1 10 23

88 kg: 15

A táblázatban csak azok a tömegek szerepeljenek, amelyeket ténylegesen el is ért Mari néni.

Beküldendő egy tömörített `i516.zip` állományban a program forráskódja és rövid dokumentációja, amely megadja, hogy a forrásállomány melyik fejlesztői környezetben fordítható.

**I/S. 46.** A 2020-as kanizsai CEOI versenyen  $N$  versenyzőnek kellett 6 feladatot megoldania. Mindegyik versenyzőről tudjuk, hogy melyik feladatra hány pontot szerzett. Adjuk meg a legkisebb különbséget, ami két versenyző összesített pontszáma közt előfordult.

*Bemenet:* az első sor tartalmazza az  $N$  számot. A következő  $N$  sor mindegyike 6 számot tartalmaz: az  $i$ -edik sor az  $i$ -edik versenyző kapott pontszámait tartalmazza sorrendben a hat feladatra.

*Kimenet:* az egyetlen sorban szerepel a keresett legkisebb különbség.

*Példa:*

Bemenet (a / jel sortörést helyettesít)	Kimenet
3 / 100 100 45 56 100 36 / 100 100 30 100 100 36 100 45 65 100 100 36	9

*Korlátok:*  $1 \leq N \leq 10^5$ . Minden pontszám 0 és 100 közti. Időkorlát: 0,2 mp.

*Értékelés:* a pontok 50%-a kapható, ha  $N \leq 100$ .

Beküldendő egy `is45.zip` tömörített állományban a megfelelően dokumentált és kommentezett forrásprogram, amely tartalmazza a megoldás lépéseit, valamint megadja, hogy a program melyik fejlesztői környezetben futtatható.

**S. 145.** Van egy szótárunk  $N$  db szóval. Azt szeretnénk tudni, hányféleképpen tudjuk a szótárból választott szavakat a dominóhoz hasonlóan összeilleszteni úgy, hogy azok  $K$  betű hosszan átfedjék egymást. Tehát a kérdés: hány olyan  $(i, j)$  rendezett számpár  $(1 \leq i, j \leq N)$  van, melyre az  $i$ -edik szó utolsó  $K$  betűjéből alkotott sorozat megegyezik a  $j$ -edik szó első  $K$  betűjéből alkotott sorozattal.

*Bemenet:* az első sor tartalmazza az  $N$  és  $K$  számokat. A következő  $N$  sor mindegyike egy, az angol ABC kisbetűiből álló (nem feltétlenül értelmes) szót tartalmaz.

*Kimenet:* a megfelelő összeillesztések, vagyis számpárok száma.

*Példa:*

Bemenet (a / jel sortörést helyettesít)	Kimenet
4 2 / sapka / kalap / baba / bamba	3

*Korlátok:*  $2 \leq N \leq 10^5$ ,  $1 \leq K \leq 100$ , minden szó legalább  $K$  és legfeljebb 100 betű hosszú. Időkorlát: 1 mp.

*Értékelés:* a pontok 30%-a kapható  $K = 1$  esetén. A pontok további 30%-a kapható, ha  $N \leq 100$ .

Beküldendő egy `s145.zip` tömörített állományban a megfelelően dokumentált és kommentezett forrásprogram, amely tartalmazza a megoldás lépéseit, valamint megadja, hogy a program melyik fejlesztői környezetben futtatható.



A feladatok megoldásai regisztráció után a következő címen tölthetők fel:

<https://www.komal.hu/munkafuzet>

**Beküldési határidő: 2020. október 15.**

