

7. feladat: Zanza (30 pont)

Tekintsük az alábbi szójátékot. Adott  $N$  darab, azonos hosszúságú szó és egy  $M$  szám. A játék célja kialakítani olyan, pontosan  $M$  betűt tartalmazó, úgynevezett zanzaszót, amelyben a megadott szavak előfordulási száma a lehető legnagyobb. Az előforduló szavak átfedhetik egymást és egy szó többször is előfordulhat a zanzaszóban.

**Feladat**

Írj programot (ZANZA.PAS vagy ZANZA.KI), amely

- A.** kiszámítja azt a  $K$  értéket, amelyhez van olyan  $M$  hosszú zanzaszó, amelyben a megadott szavak előfordulásainak száma  $K$ , és ez maximális, tehát nincs olyan  $M$  hosszú zanzaszó, amelyben  $K$ -nál több előfordulása lenne a megadott szavaknak;
- B.** megad egy olyan  $M$  hosszú zanzaszót, amelyben a megadott szavak előfordulási száma a lehető legnagyobb.

**Bemenet**

A ZANZA.BE szöveges állomány első sorában három szám van,  $N$ ,  $L$  és  $M$ .  $N$  a felhasználható szavak száma, ( $0 < N \leq 100$ ),  $L$  ( $0 < L \leq 255$ ) a szavak hossza,  $M$  ( $0 < M \leq 255$ ) pedig a kialakítandó zanzaszó hossza. A következő  $M$  sorban egy-egy  $L$  betűből álló szó található.

**Kimenet**

A ZANZA.KI szöveges állomány két sort tartalmazzon. A második sorba egy olyan  $M$  betűből álló zanzaszót kell írni, amelyben a bemenetben megadott szavak előfordulási száma a lehető legnagyobb ( $K$ ). Az előfordulások  $K$  számát az első sorba kell írni.

Példa:

ZANZA.BE	ZANZA.KI
7 4 16	8
vele	velemelvelemelve
elve	
fele	
kefe	
elem	
emel	
leve	

**8. feladat:** Domborzati térkép (40 pont)

Egy csupa nem negatív egészekből álló mátrix egy rácsnégyzetekre osztott sziget domborzati adatait tartalmazza: egy-egy érték a neki megfeleltetett rácsnégyzet fölötti, egészre kerekített, átlagos tengerszintfeletti magasságot jelenti. A domborzat viszonylag egyenletes: nem fordul elő az, hogy egy  $2 \times 2$ -es rész egyik átlójában lévő két érték kisebb a másik két értéknél. Adott koordinátájú rácsnégyzeten egy kiapadhatatlan forrás tört fel.

**Feladat:**

Írj programot (TERKEP.PAS vagy TERKEP.C), amely megadja, hogy mekkora mennyiségű vizet képesek a szigeten a keletkező tavak megtartani, illetve a tavaknak hány különböző szintje alakul ki!

**Bemenet:**

A TERKEP.BE állomány első sorában a térkép sorai ( $1 \leq N \leq 100$ ) és oszlopai ( $1 \leq M \leq 100$ ) száma van, egyetlen szóközzel elválasztva. A következő  $N$  sor mindegyike pontosan  $M$  számot tartalmaz, az egyes pontokban mért tengerszint feletti magasságot, szóközzel elválasztva. Az utolsó sorban a forrás helyének sor- ( $1 \leq K \leq N$ ) és oszlopindexe ( $1 \leq L \leq M$ ) van, egyetlen szóközzel elválasztva.

**Kimenet:**

A TERKEP.KI állomány első sorába megtartott víz mennyiségét, a másodikba pedig a tavak különböző szintjei számát kell írni.

**Példa:**

TERKEP.BE

```
7 7
0 0 0 0 0 0 0
0 20 29 29 6 53 0
0 25 25 30 16 16 0
0 41 18 36 50 21 0
0 46 9 37 60 13 0
0 33 33 33 33 33 0
0 0 0 0 0 0 0
3 3
```

TERKEP.KI

```
23
1
```

Megjegyzés: a 18 és a 9 értékű elem helyére kerül 25 az elárasztás során.

**9. feladat:** Robot (30 pont)

Egy  $2 \times 2$ -es ( $K \times K$ -as) robotot helyezünk egy szobába, megjelölve a kezdő- és a célpozícióját. A robotot el kell vezetni a célig, feltéve, hogy minden lépés 1 időegységbe kerül (csak vízszintesen vagy függőlegesen mozdulhat el 1 egységnyi), valamint, hogy úgy kell mozgatni, hogy bizonyos tárgyakat ( $1 \times 1$ -es méretűeket) felemelhet és maga mögé tehet. Egy útban álló tárgy elmozdítása szintén 1 időegységbe kerül.

**Feladat:**

Készíts programot (ROBOT.PAS vagy ROBOT.C), amely megadja, hogy minimum mennyi idő múlva érhet a robot a kezdőpozícióból a célpozícióba!

**Bemenet:**

A ROBOT.BE állomány első sorában a négyzetrács magassága és szélessége ( $1 \leq N$ ,  $M \leq 100$ ) van. A következő  $N$  sor mindegyike pontosan  $M$  karaktert tartalmaz, az üres helyeken '.'-ot, a foglalt helyeken '\*'-ot, az elmozdítható tárgyak helyén '+' jelet, a robot kezdőpozícióján négy darab 'K' betűt, célpozícióján pedig négy darab 'C' betűt.

**Kimenet:**

A ROBOT.KI állományba egyetlen sorába azt a minimális időtartamot kell írni, ami alatt a robot a kezdőhelyről a célpozícióba érhet.

**Példa:**

ROBOT.BE	ROBOT.KI
6 10	9
.....	
.KK.....	
.KK.....	
+.*****.	
.....CC..	
.....CC..	