

7. feladat: Képtár (30 pont)

A piripócsi kastély termeiben egy képkiállítást rendeztek be. N terem falain helyeztek el képeket. Egy bejárési útvonalat kell tervezni, amely az 1. teremből indul és az összes termen végighalad úgy, hogy a látogatók az összes képet pontosan egyszer láthassák, majd az 1. teremben ér véget. Ha egy teremből egy szomszéd terembe átmegyünk, akkor csak a jobb oldalon, a bejárat és a kijárat közötti falon levő képeket látja a látogató.

Feladat:

Készíts programot (KEPTAR.PAS vagy KEPTAR.C), amely az 1. teremből kiindulva megadja a bejárési útvonalat!

Bemenet:

A KEPTAR.BE állomány első sorában a termék száma ($1 \leq N \leq 1000$) van. A következő N sor az egyes termekkel szomszédos termék sorszámát tartalmazza (legalább 1, legfeljebb 20), az óramutató járásával ellentétes sorrendben. Minden sorban annyi szám van, egy-egy szóközzel elválasztva, ahány teremmel szomszédos az illető terem.

Kimenet:

A KEPTAR.KI állományba egyetlen sort kell írni, a bejárás sorrendjében a bejárás során érintett termék sorszámát.

Példa:

KEPTAR.BE	KEPTAR.KI
5	1 2 3 4 3 2 1 5 1
2 3 4 5	
1 3	
1 4 2	
1 3	
1	

8. feladat: Autó (40 pont)

Egy autót helyezünk egy labirintusba, megjelölve a kezdő- és a célpozícióját. Az autót el kell vezetni a célig, feltéve, hogy sebessége legfeljebb K lehet, s minden lépés után a sebessége 1-gyel változhat. Az autó egy időegység alatt csak vízszintesen vagy csak függőlegesen mozoghat, az S sebességű autó pontosan S távolságra lép. Az autó a kezdőpozícióban 0 sebességű (a lépése előtt 1 sebességűre vált) és a végpozícióba érve 1 sebességűnek kell lennie.

Feladat:

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

Bemenet:

Az AUTO.BE állomány első sorában a négyzettrács magassága és szélessége ($1 \leq N, M \leq 100$), valamint az autók maximális sebessége ($1 \leq K \leq 10$) van, egy-egy szóközzel elválasztva. A következő N sor mindegyike pontosan M karaktert tartalmaz, az üres helyeken szóközt, a foglalt helyeken *-ot, az autó kezdőpozícióján K betűt, célpozícióján pedig C betűt.

Kimenet:

Az AUTO.KI állományba egyetlen sort kell írni, azt a minimális időtartamot, ami alatt az autó a kezdőhelyről a célpozícióba érhet. Ha az autó nem érhet el a célpozícióba, akkor az állományba -1-et kell írni.

Példa:

```

AUTO.BE                                AUTO.KI
5 10 3                                  7
*****
*K                                     *
***** **
C
**          **

```

9. feladat: Minta (30 pont)

Szövegelemzés során gyakran kíváncsiak bizonyos szókapcsolatok előfordulására. Erre példa ez a feladat is. Adott az S szöveg, továbbá két, A és B szó. Azt szeretnénk megtudni, hogy az S szövegben hány olyan szó van, amely AWB alakú, ahol W tetszőleges (esetleg üres) szó.

Feladat:

Írj programot (MINTA.PAS vagy MINTA.C), amely meghatározza, hogy az S szövegben hányszor fordul elő AWB alakú szó, továbbá meghatározza a leghosszabb ilyen szó előfordulásának kezdő pozícióját és hosszát.

Bemenet:

A MINTA.BE szöveges állomány két sort tartalmaz. Az első sorban az A , a másodikban a B szó található. A szavak legfeljebb 100 karakterből állnak.

A SZOVEG.BE szöveges állomány egyetlen sorában van a szöveg, amelyben keresni kell. Az A és B szavak, valamint a szöveg csak az angol ábécé kis betűit tartalmazhatják. A SZOVEG.BE állományban legfeljebb 1000000 karakter van.

Kimenet:

A MINTA.KI szöveges állomány első sorába a szövegben található AWB alakú szavak számát kell írni, ha nincs ilyen, akkor 0-át. Ha az első sor tartalma nem 0, akkor a második sor két számot tartalmazzon, a leghosszabb AWB alakú szó előfordulásának kezdő pozícióját és a szó hosszát.

A minták előfordulásainak számításakor megengedett az átfedés is. A szöveg első betűjének pozíciója 1.

Példa:

MINTA.BE	SZOVEG.BE	MINTA.KI
ab	xyaabababaaabaab	6
ba		4 11