

Bergengócia királya elérkezettnek látja az időt, hogy férjhez adja leányát, ezért legjobb építészével egy sárkányokkal teli labirintust terveztet. A királylány kezét csak az a lovag nyerheti el, aki ennek egyik bejáratán belépve sikeresen kijut bármelyik másik bejáraton. A király parancsa szerint olyan labirintust kell tervezni, hogy a lovag bármelyik kijáratot is választja, útközben legalább bizonyos számú sárkányfejet le kell vágnia.

Ilyen fontos döntés esetén a király még leghűségesebb embereiben sem bízhat. Így minket kért meg, hogy írjunk programot, mely ellenőrzi, hogy az építész által elkészített labirintusterv teljesíti-e a kívánalmakat.

A program az alaprajzot fájlból olvassa, az eredményt fájlba írja. A bemeneti, illetve kimeneti fájlok nevei az első, illetve második parancssori argumentumok.

A bemenet első sorában szóközzel elválasztva  $W$ ,  $H$  és  $F$ , a labirintus szélessége, magassága és a levágandó sárkányfejek minimális  $F$  száma szerepel ( $3 \leq W, H \leq 100$ ). Az ezt követő  $H$  sor mindegyike  $W$  karaktert tartalmaz, mely:

- szóköz, ha az adott mező folyosó, melyről tetszőleges szomszédos folyosóra lehet lépni
- J, B, F és L, ha az adott mező folyosó, és egy olyan kapu van rajta, melynek hatására a mezőről jobbra, balra, fel vagy lefelé **nem** lehet lépni
- \*, ha a mező fal
- $N = 1 - 9$  számjegy, ha a mező folyosó és rajta egy  $N$ -fejű sárkány rejtőzik, melyet a mezőre lépve mindenképp le kell győzni.

Bejáratnak a labirintus szélén lévő folyosókat tekintjük, amelyekről feltehetjük, hogy mindig üresek lesznek, illetve legalább 1 fal határolja őket.

A kimenet egyetlen sora  $K A B$  alakban három, szóközzel elválasztott számot tartalmazzon: ha a terv helyes, mindhárom érték legyen 0, egyébként egy olyan  $A$  bejáratról  $B$  kijáratig vezető útvonal leírása, mely során csak  $K$  ( $K < F$ ) sárkányfejet kell levágnunk. A bejáratokat a bal felső saroktól kezdődően az óramutató járásával megegyezően, 1-től kezdve számozzuk, több megoldás esetén a legkisebb  $K$ , majd ezen belül a legkisebb  $A$ , majd  $B$  értékűt írjuk ki.

Példa be	Példa ki	
<pre> 9 6 3 ** ** *** ** **1***    3 J 1 ** **1*** ** *3 *** ** ** *** </pre>	<pre> 2 2 4 </pre>	

Beküldendő a program forráskódja (`s34.pas`, `s34.cpp`, ...), valamint a program rövid dokumentációja (`s34.txt`, `s34.pdf`, ...), amely tartalmazza a megoldás vázlatos leírását, és megadja, hogy a forrásállomány melyik fejlesztői környezetben fordítható.