

1 feladat – arrows

100 pont

"Arrows" egy játék, amelyet egy téglalap alakú táblán játszanak, amelynek a felülete $N \times M$ cellára van felosztva, amelyek N sorban és M oszlopban helyezkednek el. Mindegyik cellában egy nyíl található (fel, le, bal vagy jobb), mint a mellékelt ábrán.

←	→	→	↓	↑
→	↑	↓	←	→
↓	↑	→	→	↓
→	↑	←	←	←
←	→	↓	↓	↓
↑	↑	←	↓	↑

Mikor rá kerül a sor, egy játékos választhat egy kezdő pozíciót, amelyre zsetont helyez, majd áthelyezi a zsetont a szomszédos cellára a nyíl irányát követve. A zseton mozgatása folytatódik amíg az elhagyja a játéktáblát, és így a játékos annyi pontot kap, ahány cellán áthalad a zsetonjával.

Léteznek azonban előnyös kezdő pozíciók, amelyekből kiindulva a zseton **nem** fogja sohasem elhagyni a játéktáblát. Például mindegyik szürke háttérű pozíció előnyös. Az a játékos, aki előnyös kezdő pozíciót választ, elér egy olyan pontozást, amely a különböző egyszer látogatott cellák számának 1000-rel való szorzását jelenti.

Feladat

Ismervén a játéktábla alakját, írjatok egy programot, amely megoldja a következő elvárásokat:

1. meghatározza egy játékos által elért pontokat, aki a zsetonját egy meghatározott kezdő pozícióba helyezi;
2. meghatározza a játéktáblán levő előnyös cellák számát;
3. meghatározza a maximális pontszámot, amit a játékos elérhet ha rajta van a sor, megfelelően választva ki a kezdő pozíciót.

Bemenő adatok

Az `arrows.in` bemeneti állomány első sorában tartalmazza azt a követelményt amit meg kell oldani (1, 2 vagy 3). A második sorban az N M természetes számok találhatók, amelyek a játéktáblán levő sorok és oszlopok számát jelölik. A következő N sorban M szám található az $\{1, 2, 3, 4\}$ halmazból, amelyek a játéktábla celláiban levő nyilakat jelölik (1 jobbra mutató nyíl, 2 felfelé mutató nyíl, 3 balra mutató nyíl és 4 lefelé mutató nyíl jelöl). Az utolsó sorba a `lin col` természetes számok találhatók, amelyek a megadott kezdő pozíció sorát és oszlopát jelölik. A bemenő állomány ugyanabba a sorába írt értékek szóközzel vannak elválasztva.

Kimenő adatok

Az `arrows.out` kimeneti állomány egyetlen sort fog tartalmazni, amelybe azt a természetes számot írjuk, amely a bemeneti állomány első sorában levő követelményre adott választ jelenti.

Megszorítások és pontosítások

- $1 \leq N, M \leq 500$
- A sorok 1-től N -ig vannak számozva, az oszlopok pedig 1-től M -ig.
- **Pontszám.** A 20 pont értékű teszteknel a követelmény az 1-es. A 40 pont értékű teszteknel a követelmény a 2-es. A többi tesztnek, amelyek szintén 40 pont értékűek, a követelmény a 3-as.

Pildák

arrows.in	arrows.out	Magyarázatok
1 6 5 3 1 1 4 2 1 2 4 3 1 4 2 1 1 4 1 2 3 3 3 3 1 4 4 4 2 2 3 4 2 5 5	2000	A pldá megfelel az ábrán levő játéktáblának. Az egyes pozícióknak megfelelő pontszámok: 1 14000 14000 14000 1 15000 14000 14000 14000 1 16000 14000 14000 14000 14000 15000 14000 14000 14000 14000 1 4000 4000 2 2000 2 4000 4000 1 2000 Az 1-es követelmény: az 5-ös sorban és 5-ös oszlopban levő kezdő pozícióból indulva az elért potszám 2000.
2 6 5 3 1 1 4 2 1 2 4 3 1 4 2 1 1 4 1 2 3 3 3 3 1 4 4 4 2 2 3 4 2 5 5	23	A 2-es követelmény: 23 előnyös cella létezik.
3 6 5 3 1 1 4 2 1 2 4 3 1 4 2 1 1 4 1 2 3 3 3 3 1 4 4 4 2 2 3 4 2 5 5	16000	A 3-as követelmény: a maximális pontszámot úgy érhetjük el, ha a zsetont a 3-dik sorba és 1-es oszlopban levő kezdő pozícióba tesszük.

Maximális futási idő: 0 . 5 másodperc/teszt

Rendelkezésre álló memória: 5 MB, amelyből 1 MB a veremnek

A forráskód maximális mérete: 10 KB