

Programozás Alapjai 9. házi feladat

1. feladatsor

Szoftverfejlesztés Tanszék

2021, Ősz

Általános információk

A programot C nyelven kell megírni, és a *Bíró* webes felületén keresztül lehet benyújtani. Egy C program kiterjesztése `c`. A *Bíró* a fájl nevében található első pont utáni részt tekinti kiterjesztésnek.

Kiértékelés

A programot a *Bíró* fogja kiértékelni. Feltöltés után a *Bíró* a programot a `gcc` fordítóval és a `-O2 -static -o feladat feladat.c` paraméterezéssel lefordítja, majd a programot különböző tesztesetekre futtatja. Minden helyes teszteset 1 pontot ér. A teszteset akkor helyes, ha a program futása nem tartott tovább 5 másodpercnél, a futása hiba nélkül (0 hibakóddal) fejeződött be és az adott inputhoz tartozó kimenet **minden egyes karaktere** megegyezik az előre eltárolt referencia megoldással.

A *Bíró* által a `riport.txt`-ben visszaadott lehetséges hibakódok:

Futási hiba: 6	Memória- vagy időkorlát túllépés.
Futási hiba: 8	Lebegőpontos hiba, például nullával való osztás.
Futási hiba: 11	Memória-hozzáférési probléma, pl. tömb-túlinde克斯, null pointer használat.

Minden programra vonatkozó követelmények

A program bemenő adatait a `be.txt` nevű fájlból kell beolvasni, az eredményt pedig a `ki.txt` nevű fájlba kell írni akkor is, ha ez nincs külön megemlítve a feladat leírásában. A `be.txt` állomány csak olvasásra, a `ki.txt` állomány pedig csak írásra nyitható meg, más megnyitási mód esetén a *Bíró* nem engedélyezi a hozzáférést. Más fájl megnyitását a *Bíró* szintén nem engedélyezi.

A program bemenet/kimenet leírásokban a „sor” egy olyan karaktersorozat jelöl, amelyben pontosan egy sorvége jel (`'\n'`) található, és az az utolsó karakter. Tehát minden sort sorvége jel zár! Elképzelhető olyan output, amelyben nincs sorvége jel, de akkor a feladat kiírásának egyértelműen jeleznie kell, hogy a sorvége jel hiányzik!

A hibakód nélküli befejezést a `main` függvény végén végrehajtott `return 0;` utasítás biztosíthatja.

1. feladat: Mátrix bejárás csigavonalban (10 pont)

Adott egy mátrix, és egy kiindulási eleme. A feladat az, hogy ettől az elemtől elindulva, csigavonalban járjuk be a mátrixot, és az érintett elemeket írjuk ki. A kezdő elemtől felfelé induljunk el, majd ahol lehet balra fordulni anélkül, hogy egy elemet másodszor is érintenénk, tegyük meg, egyébként egyenesen (szokásos csigavonalban) haladjunk tovább. Ha kilépünk a mátrixból akkor is tovább kell haladni, mintha még a mátrix területén lennénk, de ekkor nem kell kiírni semmit.

Bemenet

Az első sor tartalmazza a kezdő elem koordinátáit (*sor, oszlop*) vesszővel elválasztva. A kezdő koordinátának nem kell a mátrix területére esni. Minden sort újsor karakter zár. Ezután következik egy mátrix. A mátrix mérete legfeljebb 20×20 -as lehet, de a bemenetben a méret nincs megadva előre. A bal felső eleme az 1,1 koordinátájú elem, és jobbra illetve lefelé haladva nőnek az oszlop illetve a sor koordináták. A mátrixnak kizárólag pozitív egész elemei vannak, de nem feltétlenül négyzetes alakú, azaz a sorainak és oszlopainak száma nem mindig egyezik meg.

Az inputban nincsenek whitespace karakterek (azaz szóköz, tabulátor, stb.), a számokat ' ',' (vessző) választja el egymástól, a sorban utolsó szám után nincs vessző. (Természetesen a mátrix minden sora a bemeneti fájlban is külön sorban van). A mátrixnak akkor van vége, ha a sorban egyetlen '-' jel van (természetesen a sorvége jel itt sem marad el). A **be.txt** fájl egyetlen, helyes inputot tartalmaz.

Kimenet

A mátrix elemeinek csigavonalban történő felsorolása, a **ki.txt** nevű fájlba. Az elemek egy sorban legyenek felsorolva vesszővel (',') elválasztva. A sor végét újsor karakter zárja (a kimenetben akkor is ott kell legyen egy üres sor, ha üres volt a mátrix).

Példák

1. példa

Input

```
2,3
2,1
3,0
-
```

Output

```
1,0,2,3
. ← . ← . ← .
↓   ↑
2   1 ← .   .
↓   ↓   ↑   ↑
3   0   .   .
    ↓   ↑
    . → . → .
```

2. példa

Input

```
2,2
2,1,3
3,0,4
23,12,6
-
```

Output

```
0,1,2,3,23,12,6,4,3
2 ← 1   3
↓   ↑   ↑
3   0   4
↓   ↑
23 → 12 → 6
```