

Tartalomjegyzék

- 1 Feladatok
 - ◆ 1.1 write_csv
 - ◆ 1.2 read_csv
 - ◆ 1.3
Parancssorból
futtatás
 - ◆ 1.4 Dinamikus
programozás
 - ◇ 1.4.1
Legnagyobb
közös
osztó
 - ◇ 1.4.2
Zárt
terület
kifestése

Feladatok

write_csv

Az előadáson tanultak alapján írjunk **első.csv** és **második.csv** fájlokat.

Az **első.csv**-be írjátok bele vesszővel elválasztva az első sorba a vezetéknév- és keresztnévket.

A második sorba pedig vesszővel elválasztva a kedvenc filmek címet, hogy mikor készült és a filmből a színészek a nevét, illetve a rendezőt.

A **második.csv** fájl legyen hasonló az **első.csv**-hez: Első sorba írjátok bele tabulátorral elválasztva a vezetéknév- és keresztnévket.

A második sorba írjátok bele, hogy a matematika mely területe érdekel titeket a legjobban.

A harmadik sorban pedig az szerepeljen, hogy mit dolgoznátok szívesen a jövőben. (Több dolgot is fel lehet sorolni.)

Például az én esetemben így néznének ki ezek a fájlok:

első.csv:

Csonka,Bence

Aljas nyolcas,2015,Samuel L. Jackson

második.csv:

Csonka Bence

Kombinatorika

Ács Filozófus

read_csv

Írjunk egy függvényt, ami beolvasson egy .csv fájlt.

A függvénynek legyen egy opcionális delimiter paramétere, aminek a default értéke legyen ','.

Írjuk meg úgy, hogy működjön a függvény akkor is, ha nem ','-vel vannak elválasztva az értékek a fájlban, hanem tabulátorral. (Ebben az esetben a bemenetben indikálja ezt delimiter='\t' paraméter is, nem kell, hogy automatikusan felismerje.)

A függvény adja vissza listák listáját, amiben a listák a soroknak felelnek meg, az elemek pedig a sorban található szavak megfelelően szétválasztva.

Parancssorból futtatás

Írjunk egy `prim.py` nevű programot, amit parancssorból tudunk futtatni, a bemenete tetszőlegesen sok pozitív egész szám és térjen vissza ezek legnagyobb közös osztójával. Emeljünk kivételt, ha valamelyik nem egy pozitív egész szám.

Tipp: Ehhez használjuk a `sys` csomagot.

Dinamikus programozás

Legnagyobb közös osztó

Implementáljuk a legnagyobb közös osztó függvényt (`lnko`) az euklideszi algoritmus segítségével rekurzívan!

Zárt terület kifestése

Olvassuk be a `picture.txt` fájlt listák listájába (minden karakter egy elem)! Írjunk egy `fill(x,y)` függvényt, ami ugyanazt csinálja, mint a Paint kitöltés funkciója! Az (x,y) pontból kiindulva a `.` helyére `#` jelet tesz, amíg a `#` jel által jelölt falba nem ütközik! A módszer rekurzív: kifestjük az (x,y) pontot, majd a szomszédait, ha azok nem `#` jelek. Hívjuk meg a szomszédokra (akik nem `#` jelek) a függvényt rekurzívan. Ha nincs kit kiszínezni, akkor álljunk meg!

```
.....
..#####.....
..#.....#.....
..#.....#.....
..#.....#.....
..#.....#.....
..#.....#.....
..#.....#####.....
..###.....##.....#.....
..#..##.....##.....#.....
..#...##.....##.....#.....
..#.....##.....#.....
..#.....#.....#.....
..#.....#.....#.....
..#.....##.....#.....
..#.....##.....#.....
..#.....##.....#.....
..#####.....
.....
.....
.....
.....
```