

# Szkriptnyelvek

## – 1. ZH –

2021. ápr. 1.

### Feladatok

#### 1. (5 pont) Összefésülés

Írjon egy programot, ami parancssori argumentumként vár három sztringet. A program fésülje össze a három sztringet s írja ki a képernyőre az eredményt.

Összefésülni csak azonos hosszúságú sztringeket lehet. Éppen ezért ha egy sztring rövidebb mint a leghosszabb sztring, akkor a sztring végét 'X' karakterekkel kell feltölteni. Például ha az „ab”, „cde” és „fghi” sztringeket adjuk meg, akkor a program az „abXX”, „cdeX” és „fghi” sztringeket fogja összefésülni.

Az összefésülés alatt azt értjük, hogy először vesszük mindegyik sztring legelső karakterét. Majd vesszük a sztringek második karakterét, stb.

Például az „ab”, „cde” és „fghi” összefésülése az „acfbdgXehXXi” eredményt adja.

Ha a programnak nem pontosan három parancssori argumentumot adunk meg, akkor a program álljon le egy informatív hibaüzenettel. (Ha az ékezetek gondot okoznak, akkor az ékezetek használatától el lehet tekinteni).

Futási példák:

```
$ ./prg.py
Hiba! Adj meg pontosan három sztringet!
```

```
$ ./prg.py aa
Hiba! Adj meg pontosan három sztringet!
```

```
$ ./prg.py aa bb
Hiba! Adj meg pontosan három sztringet!
```

```
$ ./prg.py aa bb cc dd
Hiba! Adj meg pontosan három sztringet!
```

```
$ ./prg.py lac eva ana
leaavncaa
```

```
$ ./prg.py Laci eva Anna
LeAavncaniXa
```

## 2. (5 pont) A pizzafutár

### A) változat (részpont: 3 pont)

Pizzafutárként vállaltunk nyári munkát, de a legutóbbi kiszállítással meggyúlt a bajunk. Egy toronyházba kell kivinni a pizzát, de az emeletet egy kicsit furcsán adták meg. Kezdetben a földszinten vagyunk (0. emelet), s egy zárójeleket tartalmazó sztring adja meg, hogy hova kell mennünk. A nyitó zárójel jelentése: menjünk egy emeletet fel. A csukó zárójel jelentése: menjünk egy emeletet le.

Például:

- Mind a `(( ))`, mind a `() ()` jelentése: a cél a 0. emelet.
- Mind a `(( (`, mind a `(( () () (` jelentése: a cél a 3. emelet.
- A `) ) ( ( ( ( (` jelentése: a cél a 3. emelet.
- Mind a `) ) )`, mind a `) () () ()` jelentése: a cél a -3. emelet (a földszint alatt kiterjedt alagsorrendszer található).

A cél emelethez vezető instrukciók az `input.txt` állományban találhatók.

Írjon egy programot, ami feldolgozza az `input` file tartalmát, s kiírja a képernyőre, hogy melyik emeletre kell kiszállítani a pizzát. A program kimenete a következőképpen nézzen ki:

```
$ ./prg.py
Cél emelet: 25
```

A „25” helyett természetesen a helyes eredményt kell feltüntetni.

### B) változat (részpont: 2 pont)

Bővítsük ki az előbbi programot. Tegyük fel, hogy az `input.txt` állományban található utasításokat egyenként hajtjuk végre. A bolyongásunk során melyik az a legmagasabb helyen lévő emelet, ahova felmentünk? Illetve melyik az a legmélyebben lévő alagsori szint, ahova lementünk?

Például:

- A `) ((( ( ( ( ( ( ( ( ( (` instrukciókat követve feljutottunk a 3. emeletre (legmagasabb pont), illetve lementünk a -2. szintre (legmélyebb pont).

Bővítsük ki az A) verziót, s a programunk most már ezt a két plusz információt is írja ki.

A program kimenete a következőképpen nézzen ki:

```
$ ./prg.py
Cél emelet: 25
Legmagasabb emelet: 56
Legmélyebb emelet: -20
```

Természetesen a helyes értékeket kell majd feltüntetni.

### C) változat (szorgalmi +1 extra pontért) [ezt a legvégén oldják meg, ha még maradt idő]

Bolyongásunk során melyik emeletet érintjük a legtöbbször? Amikor belépünk az épületbe, akkor ezt úgy vesszük, hogy a 0. emeletet már egyszer érintettük. Például a `(( ( ( ( ( ( ( ( ( (` hatására felmegyünk a 2. szintre, utána lemegyünk a -2. szintre, majd vissza a földszintre. Így a földszintet érintjük a legtöbbször (háromszor).

```
$ ./prg.py
...
Legmélyebb emelet: -20
Legtöbbször érintett emelet: 3
```

### 3. (5 pont) Napi bölcsesség

Adott egy szöveges állomány. Minden sorban egy egész szám és egy sztring található, amik egymástól pontosvesszővel vannak elválasztva.

A sztring a pontosvessző kivételével bármilyen karaktert tartalmazhat.

Példa:

```
25;tiobe
8;java
16;elem
3;lua
0;ssh
4;c++
15;python
9;ruby
1;elem
```

Az állományban egy titkos üzenet van elrejtve, amihez a következőképpen tudunk hozzáférni: vegyük azokat a sorokat, ahol a szám négyzetszám. Rendezzük ezen sorokat a négyzetszám szerint, majd vegyük a szavak első betűjét. A fenti példában a „secret” szó lett ezzel a módszerrel elrejtve.

A mellékelt `input.txt` állományban egy Coelho-bölcsesség lett elrejtve. Vajon mi lehet ez?

A program futtatása a fenti példára és az elvárt kimenet formátuma:

```
$ ./prg.py
secret
```

A programot a mellékelt `input.txt` állományra futtassa!

A kimeneten még végezzen el egy apró finomítást: az aláhúzásjeleket cserélje le szóközökre!

4. (3 pont) **Fibó**

A Fibonacci-sorozat következő elemét úgy kapjuk meg, hogy vesszük a sorozat előző két elemének az összegét. Ha a sorozat az 1 és 2 értékekkel indul, akkor a sorozat első 10 eleme a következő lesz:

1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

Mennyi lesz a sorozat 4 milliónál kisebb *páros* elemeinek az összege?

A program kimenete egyetlen egész szám (a fenti kérdésre adott válasz) legyen.

Oldalszám:	1	2	3	4	Összesen
Max. pontszám:	5	5	5	3	18
Pontszám:					