

# Eltérés

Készítsünk függvényt `elteres` néven, amely három számot kap paraméterként, és kiszámítja majd visszaadja az első számnak a másik két számtól való abszolút eltérése közül a nagyobbbat! Például `elteres(3, 5, 10)` függvényhívás esetén a 3 és 5 eltérése 2, 3 és 10 eltérése 7, így a visszaadott érték 7. Vagy `elteres(6, 1, 3)` függvényhívás mellett 6 és 1 eltérése 5, 6 és 3 eltérése 3, tehát visszaadja az 5-öt.

Készítsen főprogramot `elter.py` néven, amely bekér a függvényben szereplő három számot, majd meghívja a függvényt, eltárolja a kapott eredményt és kiírja azt.

Minta:

```
Első szám: 6
Második száma: 2
Harmadik szám: 9
Az eltérés: 4
```

# Szövegel

Készítsünk függvényt `szovegel` néven, amely egy szöveget és egy egész számot kap bemenetként, majd visszaszadja a szöveg paraméterként kapott szám-adik karakterét, vagy egy “\_” jelet, ha szám-adik karaktere nincs a szövegnek. Például `szovegel("ablak alatt", 3)` függvényhívás esetén a visszaadott érték legyen az el betű: “l”. Vagy `szovegel("ablak alatt", 17)` függvényhívás mellett a visszaadott érték “\_” az aláhúzásjel.

Készítsen főprogramot `szovegelo.py` néven, amely bekéri a függvényben szereplő szöveget és számot, majd meghívja a függvényt, eltárolja a kapott eredményt és kiírja azt.

Minta:

```
Szöveg: Programozás
Száma: 5
Az eredmény: r
```

# PárosÁtlag

Készítsünk függvényt `paros_atlag` néven, amely egy egész számokból álló listát kap bemenetként, és kiszámítja majd visszaadja a listában szereplő páros számok átlagát! Ha a listában nincs páros szám, akkor adjon vissza 0-t! Például `paros_atlag({3,5,10})` függvényhívás esetén 0-t kapunk, vagy `paros_atlag([6,1,4,8])` függvényhívás mellett 6.0-t kapunk.

Készítsen főprogramot `paratlag.py` néven, amely bekér a függvényben szereplő lista számait egy sorban, szóközzel elválasztva, majd meghívja a függvényt, eltárolja a kapott eredményt és kiírja azt.

Minta:

```
Számok: 6 8 3 7 2 1
```

```
A párosak átlaga: 5.3333333
```

## Egymás után

Készítsünk függvényt `egymas` néven, amely egy szavakból álló listát kap bemenetként, és megadja, hogy a listában előfordul-e, hogy két egymás követő szó egyforma! Például `egymas(["ablak","ajto","kilincs","ajto"])` függvényhívás esetén a visszaadott érték hamis (False) míg `egymas(["ablak","ajto","ajto","kilincs"])` függvényhívás mellett a visszaadott érték igaz (True)! Figyeljünk arra, hogy a függvény üres lista bemenetre is működjön, amire természetesen hamis értéket kell adnia.

Készítsen főprogramot `egyutan.py` néven, amely bekéri a függvényben szereplő szavakat egy-egy szóközzel elválasztva, elkészíti a függvényhíváshoz szükséges listát, meghívja a függvényt, eltárolja a kapott eredményt és válaszol a kérdésre, hogy van-e két egymás követő azonos szó!

Minta:

```
Szavak: ablak ajto ajto kilincs
```

```
Van egymást követő azonos szó
```

Minta2:

```
Szavak: ablak ajto ablak kilincs
```

```
Nincs egymást követő azonos szó
```

```
Szavak: (a bemenet csak egy enter)
```

```
Nincs egymást követő azonos szó
```



## Első páratlan vagy nulla

Készítsünk függvényt `első_paratlan` néven, amely egy egész számokból álló listát kap bemenetként, és visszaadja a listában szereplő első páratlan számot, vagy 0-t, ha nincs a listában páratlan szám! Például `első_paratlan([3,5,10])` függvényhívás esetén 3-at kapunk, vagy `első_paratlan([6,1,4,8])` függvényhívás mellett 1-et kapunk, de `első_paratlan([6,4,8])` mellett 0-t kapunk.

Készítsen főprogramot `epar.py` néven, amely bekér a függvényben szereplő lista számait egy sorban, szóközzel elválasztva, majd meghívja a függvényt, eltárolja a kapott eredményt és kiírja azt.

Minta:

Számok: 6 8 3 7 2 1

Az első páratlan: 3

## Kisebb - nagyobb

Készítsünk függvényt `kisebb_nagyobb` néven, amely egy egész számokból álló listát és egy számot kap bemenetként. A függvény készítsen két listát: az egyikben a kapott számnál kisebb, a másikban a kapott számnál nagyobb számok legyenek. A függvény adja vissza a két listát egy kételemű listába csomagolva. Például `kisebb_nagyobb([3,5,10,4],5)` függvényhívás esetén a visszakapott kételemű lista `[[3,4],[10]]`, vagy `kisebb_nagyobb([6,1,4,8],20)` függvényhívás mellett `[[6,1,4,8],[]]` -et kapunk. Az második paraméterként kapott számmal egyező szám egyik listában sem szerepel.

Készítsen főprogramot `kisnagy.py` néven, amely bekér a függvényben szereplő lista számait egy sorban, szóközzel elválasztva, bekéri a számot, majd meghívja a függvényt, eltárolja a kapott eredményt és kiírja azt.

Minta:

Számok: 6 8 3 7 2 1

A szám: 4

Kisebbek: [3, 2, 1]

Nagyobbak: [6, 8, 7]

# Angol szavak 1

Az `eng5000.txt` egyszerű szöveges állományban az interneten előforduló 5000 leggyakoribb angol szó szerepel. Minden sorban egy szó, majd szóközzel utána a szó gyakorisága. Az állományban gyakoriság szerint csökkenő sorrendben vannak a szavak, tehát a leggyakoribb a legelső, a második leggyakoribb a második stb.

Készítsen programot **angol1** néven, abban oldja meg az alábbi feladatokat! A megoldás során minden esetben jelezze, hogy milyen eredményt ír ki vagy milyen inputot vár a felhasználótól. Ehhez vegye figyelembe a feladat végén lévő mintát!

Feladatok:

1. Olvassa be a szöveges állomány számait, és tárolja el egy megfelelő adatsorozatban!
2. Kérjen be egy angol szót, és adja meg, hogy szerepel-e a beolvasott szavak között, és ha szerepel, akkor hányadik, és hányszor fordul elő! Ha nem szerepel, akkor írja ki, hogy nincs a beolvasott szavak között!
3. Kérjen be egy gyakoriságot, ami az előző két gyakoriság között van, és adja meg az első szót, amelyik legalább ekkora gyakorisággal rendelkezik!
4. Készítsen függvényt `hetbetus` néven, amelynek bemenete egy szöveg, visszaadott értéke egy logikai érték, és megadja, hogy a szöveg hét karakterből áll, vagy sem! Ezt felhasználva számítsa ki a program, hogy melyik a legkisebb és legnagyobb gyakoriságú hétbetűs szó!
5. Egy nyelvész kíváncsi arra, hogy vajon az ötbetűs vagy a hatbetűs vagy hétbetűs angol szavak fordulnak-e elő többször. Adja meg, hogy hány hatbetűs szó van a leggyakoribbak között, és összesen hányszor fordulnak elő, valamint adja meg, hogy hány hétbetűs szó van, és összesen hányszor fordulnak elő!

Minta:

Kérek egy szót: `program`

Ez a(z) 199. szó, előfordul 306686983 alkalommal

Legnagyobb gyakoriság: 23135851162

Legkisebb gyakoriság: 13264592

Adjon meg egy gyakoriságot: 15000000

Az első ilyen gyakori szó: `pour`, gyakorisága: 15002685

A legkevésbé gyakori hétbetűs: `contact`

A leggyakoribb hétbetűs szó: `insider`

Hatbetűs szavak száma: 768 előfordulnak: 43860908989

Hétbetűs szavak száma: 736 előfordulnak: 40048529727