

- Készítsen `Sztringek()` néven eljárást, amelyben az alábbi feladatokat oldja meg!
 - Kérjen be a felhasználótól egy tetszőleges mondatot, amelyet egy erre alkalmas változóban tárol el! Végezze el az alábbi feladatokat!
 - Írja képernyőre, hogy hány darab szóközt tartalmazott a mondat!
 - Minden szóközt cseréljen '*' karakterre!
 - Írja képernyőre, hogy a mondatban szerepel-e a 'szép' szó!
 - Írja képernyőre a mondatot csupa nagybetűvel!
 - Jelenítse meg a mondat első és utolsó 5 karakterét!
 - Írja képernyőre, hogy '.' (pont – írásjel) van-e a mondat végén. Ha a felhasználó nem pontot használt a mondat végén, írja képernyőre a mondat utolsó karakterét!
 - Tárolja el egy változóba a 'cukor' szót, majd fűzze a felhasználó által megadott mondathoz!
 - Darabolja a szöveget szóközök mentén, majd írja képernyőre az egyes részeket!
 - Szúrja be a mondat elejére a saját nevét!
 - Törölje a mondat utolsó karakterét!
 - Adja meg a '*' karakter első és utolsó indexét, ha egyáltalán szerepel a megadott karakter a mondatban!
 - Írja képernyőre az utolsó szó első betűjét!

```

Kérek egy mondatot!
A termékek csomagolása színben és mintában eltérhet.

A mondatban 6 darab szóköz szerepel

A szóközök cseréje * karakterre: A*termékek*csomagolása*szinben*és*mintaban*eltérhet.

A mondatban nem szerepel a 'szép' szó.

A mondat nagybetűsen: A TERMÉKEK CSOMAGOLÁSA SZÍNBEN ÉS MINTÁBAN ELTÉRHET.

A mondat elő 5 karaktere: A ter
A mondat utolsó 5 karaktere: rhet.
A mondat végén . van.

Az összefűzött mondat: A termékek csomagolása színben és mintában eltérhet.cukor

A
termékek
csomagolása
színben
és
mintában
eltérhet.
Saját Név
A termékek csomagolása színben és mintában eltérhet.

A mondat az utolsó karakter nélkül: A termékek csomagolása színben és mintában eltérhet

Nincs benne '*'.

Az utolsó szó első betüje: e

```

- Készítsen egy gyümölcsöket tartalmazó listát 'gyumolcsok' néven az alábbi kezdőértékekkel: eper, kiwi, ananász, görögdinnye, citrom, gránátalma.
- Készítsen függvényt `BetuDarab()` néven, amely visszaadja a megadott nevek karaktereinek összértékét!
- Készítsen függvényt `Leghosszabb()` néven, amely meghatározza a legrövidebb nevet a listában!
- Készítsen egy `Feltoltés()` eljárást, amely egy `szamLista` névű listát tölt fel háromjegyű véletlen számokkal egészen addig, míg a lista elemeinek összege nem éri el a 2500-at! A véletlen szám csak úgy kerüljön a listába rögzítésre, hogy a lista elemeinek összege legfeljebb 2500 lehet!
- Készítsen egy `Kiir()` névű eljárást, amely megjeleníti a `szamLista` lista elemeit egymástól szóközzel elválasztva!
- Készítsen `Osszeg()` néven függvényt, amely visszaadja a lista páros elemeinek összegét!

8. Készítsen `Vizsgalat()` néven eljárást, amely megvizsgálja az `Osszeg()` függvény segítségével (felhasználásával), hogy a lista elemeinek összege kisebb-e, mint 2500?
9. Készítsen `Darab()` néven függvényt, amely visszaadja a lista elemeinek darabszámát!
10. Készítsen eljárást `ListaTolt()` néven, feltölti a `szamLista` nevű listát további 25 db 100 és 110 közötti véletlen számmal!
11. Írja képernyőre a `szamListalista` tartalmát!
12. Készítsen függvényt `HaromDarab()` néven, amely megjeleníti az hárommal osztható számok darabszámát!
13. Készítsen `BenneVan()` néven függvényt, amely eldönti, hogy a felhasználó által bekért érték szerepel-e a `szamLista` listában! Az eredményt a mintának megfelelően jelenítse meg!

```
3. feladat:  
A gyümölcsök lista összes karakterszáma: 42  
  
4. feladat:  
A lista legtöbb betűből álló gyümölcs: görögdinnye  
  
A 'szamok' lista elemei: 733 865 691  
  
7. feladat:  
A lista páros elemeinek összege: 0  
  
Az elemek összege nem éri el a 2500-at.  
  
9. feladat:  
A lista 3 db elemet tartalmaz.  
  
A 'szamok' lista elemei: 733 865 691 104 102 100 100 110 101 102 105 102 105 109 103 109 108 104 107 106 107 107 101 103 106  
100 106 107 100  
  
12. feladat:  
A listában 6 db hárommal osztható szám van.  
  
13. feladat:  
  
Melyik elemet keresi?  
888  
A keresett szám nincs a listában.
```