|  |  |  |  |
| --- | --- | --- | --- |
| Idő | Óra menete | Leírás | Comment |
| 0-5 | Köszönés, ismétlés | Gépek bekapcsolása.  Mi volt múlt órán? | Válasz: Do while, if |
| 5-15 | Ismétlés | Hozzunk létre 10 int változót. Ezek lesznek egy tanuló matek jegyei. Majd számoljuk ki az átlaga alapján, hogy hányas lesz félévkor és írassuk is ki.  **Röviden**:  int a=2,b=5,c=4,d=3,e=3,f=2,g=1,h=1,i=5,j=3;  a+=b;   a+=c;    a+=d;    a+=e;  a+=f;   a+=g;    a+=h;    a+=i;    a+=j;  double atlag = a/10.00;  Console.WriteLine(atlag); |  |
| 15-25 | Tömb | Ennél sokkal egyszerűbb, ha tömböket használunk.  int[] jegyek = {2,5,4,3,3,2,1,1,5,3};  Int után szögletes zárójel, a tömböt jelenti, majd egyenlőségjel után megadjuk a kezdőértékeit. Az értékeket benne indexelve tudjuk kivenni. A legelső eleme a 0. és így minden további eleme is 1-el el van csúsztatva.  Le kell nulláznunk az átlag változót, majd ezután egy foreach ciklussal összeadjuk a számokat.  atlag = 0; //mivel double, így olyan mintha 0.00-t írtam volna oda  foreach(double jegy in jegyek){              atlag += jegy;          }  A double helyett a jegy lehet int is, itt most lényegtelen.  A jegy változóba minden elem egyesével „meg fog jelenni”, így pontosan annyiszor fog lefutni, mint ahány elemű a jegyek tömbünk. Ezeket hozzá adjuk az atlag változónkhoz.  Console.WriteLine(atlag/jegyek.Length);  A kiíratásnál nem kell feltétlen tudnunk, hogy hány számot adtunk össze vagy hány elemű a tömbünk, mert a Length megmondja nekünk.  Mivel az atlag double típus, így nem kell konverzió a tört átlaghoz. |  |
| 25-35 | Tömb 2 | **Feladat**: Hozzunk létre egy 5 elemű tömböt, majd kérjünk be számokat és számoljuk ki a tömbben levő számok átlagát.  int[] osztalyzatok = new int[5];  atlag = 0;  for(i=0;i<osztalyzatok.Length;i++){  osztalyzatok[i] = int.Parse(Console.ReadLine());  atlag += osztalyzatok[i];  }  Console.WriteLine(atlag/osztalyzatok.Length);  int.Parse() kell ahhoz, hogy a string beolvasásból int típus legyen. Az átlag számolásához egyszerűbb, ha egyből hozzá adjuk az atlag változóhoz a beolvasott értéket, így nem kell még egy ciklus.  A for ugyan az mint a foreach csak jobban testre szabható, nem muszáj végig mennie az elemeken, illetve a tömb elemeit így indexelve tudjuk vissza kapni. |  |
| 35-55 | Gyakorlás | **Feladat**: |  |
|  |  |  |  |
|  |  |  |  |

## Az otthoni gyakorló feladatok a [4.hazi.cs](https://github.com/DrCode17/CS) file-ban elérhetőek.