**1.  
A programozásnál alkalmazott egyszerű és összetett adattípusok, ezek jellemzői, és az egyes adattípusokhoz tartozó jellemző műveletek**

Programjainkban adatokat dolgozunk fel. Az adatokat változókban tároljuk. A változók típusa függ a tárolásra kerülő adattól. A különböző típusok más-más kódolással kerülnek a memóriába, de egy típust is többféleképpen kódolhatunk. A kódolás típusától függ a tárolható értékek halmaza, azaz a típus értékhalmaza.

Az **adattípusok** az adatok szerkezetének, értékhalmazának, kódolási módjának, továbbá a hozzátartozó műveleteknek, relációknak, tulajdonságoknak és metódusoknak az összessége. Az adattípusaink értékhalmazok alapján kétfélék lehetnek: **elemiek (egyszerűek)**, amelyeknek felhasználói szempontból nincs belső szerkezetük, valamint **összetettek**, amelyek elemiekből épülnek fel; ezek az adatszerkezetek.

**Egyszerű adattípusok**

Az egyszerű adattípusok szerkezetileg nem bonthatók tovább. Ilyen pl.: egész szám, valós szám, karakter, logikai érték.

* *Egész szám:* Az egész számokat legtöbbször 2 byte-on ábrázoljuk, kettes komplemens kódban. Fixpontos ábrázolás. Lehet előjeles vagy előjel nélküli. Relációs és matematikai műveleteket tudunk végrehajtani. C#: int, byte
* *Valós szám*: Véges sok számjeggyel leírható racionális számok. Lebegőpontos ábrázolás. C#: float, double. Műveletek: értékadás, relációs és matematikai
* *Logikai érték*: 1 biten vagy byte-on ábrázoljuk. Értékhalmaza az igaz/hamis. Logikai műveletek: AND, OR, XOR, NOT. Pl.: C#: bool.
* *Karakter:* ASCII vagy UNICODE kódrendszeren ábrázoljuk. Értékhalmaza az adott kódtábához tartozó jelek halmaza. Relációs műveletek végrehajtása. Pl.: C#: char. Műveletek: értékadás, relációs műveletek.

**Összetett adattípusok**

Értékhalmaz mellett a szerkezetük is fontos. Bonyolultabb ábrázolásmód jellemzi. Ilyen pl.: tömb, rekord (struktúra), szöveg (karaktersorozat), sor, sorozat, verem.

* *Sorozat:* Azonos típusú adatok összessége lineáris rendezéssel. Műveletek: egy elem kiválasztása; **azonosító[index]**
* *Tömb:* Azonos típusú adatok összessége lineáris rendezéssel. Művelet: egy elem kiválasztása**; azonosító[index1….indexN]**
* *Rekord:* Különböző típusú, de logikailag összetartozó adatok összessége. Relációs műveletek, elem kiválasztása: **azonosító.adatazonosító.** C# példa: struktúra
* *Szöveg:* Karakter típusú adatok tömbje vagy sorozata. Műveletek: értékadás, összefűzés (+), relációs műveletek, egy elem vagy karakter kiválasztása; **azonosító[index]** . C# példa: string
* *Sor:* Azonos típusú adatok összessége. Sor műveletek: üres, üres-e, első, sorba (in), sorból (out)
* *Verem:* Azonos típusú adatok összessége. Verem műveletek: üres, üres-e, felső, verembe (push), veremből (pop).

**Tárolás szerint lehetnek:**

* homogén (mindegyik adat egyforma típusú) – pl. tömb
* inhomogén (adatok különböző típusúak) – pl. rekord
* statikus (előre ismert a darabszám)
* dinamikus (futás közben változtatható a darabszám)

**Kezelés szerint**

* véletlen elérésű (Random Access) (az adatok bármelyike egyforma sebességgel elérhető)   
  – pl. tömb, rekord
* soros elérésű (az adatok csak egymás után érhetők el) – pl. sor, verem