

Microsoft Small Basic

Rövid leírás kezdőknek

II. rész

Elágazások

Az utasításoknak egy új csoportjával ismerkedünk most meg, ezek az elágazások. Olyan ez, mint amikor egy útelágazáshoz érünk, és attól függően, hogy mi az célunk, választjuk ez egyiket vagy másikat.



Ha azt szeretnénk, hogy programunk egy feltételtől függően tegyen meg, vagy éppen ne tegyen meg valamit, akkor az adott helyen a programban elágazást kell használni!

Az elágazás utasítás általános alakja:

```
If feltétel Then  
    utasítások, amelyeket akkor kell végrehajtani, ha a feltétel igaz  
EndIf
```

Tehát az **if** (magyarul a „ha”) szó után adjuk meg a feltételt. Ebben általában valamilyen összehasonlítást végzünk. Például a számok összehasonlítása a következőképpen történik:

Reláció	Jelölése (egy példával)	Jelentése
egyenlő	$x=10$	Az x változó értéke egyenlő-e 10-el?
kisebb	$x<10$	Az x változó értéke kisebb-e 10-nél?
nagyobb	$x>10$	Az x változó értéke nagyobb-e 10-nél?
kisebb vagy egyenlő	$x\leq 10$	Az x változó értéke kisebb vagy egyenlő-e mint 10?
nagyobb vagy egyenlő	$x\geq 10$	Az x változó értéke nagyobb vagy egyenlő-e mint 10?
nem egyenlő	$x\neq 10$	Igaz-e, hogy x értéke nem 10?

Példák

Írjunk programot, amely meghatározza egy a billentyűzetről beolvasott szám abszolút értékét a következő módon: ha a szám negatív, akkor megszorozza mínusz egygel (ha nem negatív, akkor változatlanul hagyja). Megoldás:

```
TextWindow.Write("Kérem a számot: ")  
szam=TextWindow.Read()  
If (szam < 0) Then  
    szam=szam*-1  
EndIf  
TextWindow.WriteLine("Abszolút értéke: "+szam)
```

Nézzük a következő programot!

```
TextWindow.Write("Add meg, hány óra van: ")
idő= TextWindow.Read()
If (idő < 10) Then
    TextWindow.WriteLine("Jó reggelt!")
EndIf
If (idő >= 10) Then
    TextWindow.WriteLine("Jó napot!")
EndIf
```

Hogyan működik ez a program?

Először kiírja, hogy milyen adatot vár tőlünk, majd az **idő** nevű változóba beolvassa a beírt adatot. Az első elágazás megvizsgálja, hogy az adat kisebb-e 10-nél. Ha igen, akkor a „Jó reggelt!” szöveget írja ki, különben nem csinál semmit. Ezt követően a második elágazás újra megvizsgálja az adatot. Ha az adat 10 vagy annál nagyobb érték, akkor a „Jó napot!” szöveget írja ki, különben nem csinál semmit.

Egy elágazásban nem csak egy utasítás szerepelhet, hanem több is, sőt akár másik elágazás is. Ügyeljünk arra, hogy az elágazás végét jelölő EndIf jelzést mindig valóban az elágazás végére tegyük.

A különben ág: ELSE

Sok esetben az elágazásoknál hasznos, ha nem csak azt tudjuk megmondani, hogy mi történjen, ha az If utáni feltétel teljesül, hanem azt is, hogy mi történjen különben, tehát akkor, ha a feltétel nem teljesül. Nézzük az előző köszönős példánkat az Else (magyarul: „különben”) jelölés használatával!

```
TextWindow.Write("Add meg, hány óra van: ")
idő= TextWindow.Read()
If (idő < 10) Then
    TextWindow.WriteLine("Jó reggelt!")
Else
    TextWindow.WriteLine("Jó napot!")
EndIf
```

Most ugyanannak a feladatnak az elvégzésére elegendő egy elágazás használata, aminek van egy különben ága is. Az If utasítás Else ágában lévő utasításokat a program akkor hajtja végre, ha a feltétel nem teljesül. Ennek megfelelően a „Jó napot!” szöveget a program akkor írja ki, ha nem igaz az idő<10 feltétel.

Látjuk, hogy az If utasítás az eseteket két csoportra osztja. Az egyik csoport az, amelyekre teljesül a feltétel, a másik az, amelyekre nem. Ha ez utóbbi eseteket is szét kell választani, akkor az elágazás valamelyik ágába egy újabb elágazást kell beépítenünk! Nézzünk egy példát!

A feladat az, hogy eldöntsük egy beolvasott számról, hogy pozitív, negatív vagy nulla. A feladat megoldásához három eset elkülönítésére van szükség. Eljárhatunk pl. a következő módon: először különítsük el azt az esetet, amikor a szám zérus, majd a maradék esetekből válasszuk ki azokat, amelyekben a szám pozitív. Ha egy számra ez a feltétel sem igaz, akkor az csak negatív lehet. A feladat egy megoldása lehet a következő:

```
TextWindow.Write("Adj meg egy számot: ")
szam= TextWindow.Read()
If (szam < 0) Then
    TextWindow.WriteLine("A szám negatív")
else
    If (szam > 0) Then
        TextWindow.WriteLine("A szám pozitív")
    Else
        TextWindow.WriteLine("A szám NULLA")
    EndIf
EndIf
```

Gondold végig és próbáld ki, hogy miért nem lenne jó megoldás, ha a második elágazást nem az első elágazás különben ágába tennénk, hanem az első elágazást lezárva egy külön elágazást készítenénk belőle!

Kiegészítés: néhány matematikai művelet

Érdekes programok megírására adnak lehetőséget a következő matematikai műveletek.

Maradékképzés

Maradékképzés: a segítségével meg lehet állapítani, hogy egy számot egy másikkal osztva mennyi lesz a maradék.

```
Math.Remainder(x,y)
```

Megadja az x/y hányados maradékát. (A maradékos osztás maradékát.)

pl.: ha egy X szám páratlan, akkor: $\text{Math.Remainder}(x,2)=1$

Kerekítés

```
Math.Floor(x)
```

Megadja x lefelé kerekített értékét. Pl. $x=3,75$ esetén 3 az eredmény

Véletlen egész szám előállítása

```
Math.GetRandomNumber(x)
```

Véletlenszerűen kiválaszt egy egész számot 1 és x között és ezt a számot adja meg.

Példa: állítsuk elő egy kockadobás eredményét!

```
kocka=Math.GetRandomNumber(6)
```

A **kocka** változó tartalma 1 és 6 közötti egész szám lesz.

Számok abszolút értéke

```
Math.Abs(x)
```

Megadja a zárójelben kapott szám abszolút értékét.

Például:

```
Textwindow.WriteLine(Math.Abs(-8))
```

A képernyőn 8 fog megjelenni