

VI.5. Delegáltak, események

Ahogy korábban is szó volt róla, a biztonságosabb programkód készítésének érdekében a mutató aritmetika a C# nyelvben nem engedélyezett. Ebbe beletartozik az is, hogy a függvénymutatók sem lehetnek kivételek.

Ez utóbbi esetben viszont olyan nyelvi tulajdonságok nem implementálhatók, mint például egy menüponthoz hozzárendelt függvény, amit a menüpont választása esetén kell végrehajtani. Ezen utóbbi lehetőséget hivatott a delegált típus biztosítani. Ez már csak azért is fontos, mert többek között a Windows programozás „*callback*” jellemző paramétere is ezt használja.

A C++ környezetben ezt a lehetőséget a függvénymutató biztosítja.

A delegált valójában egy függvénytípus-definíció, aminek alakja a következő:

`delegate típus delegáltnév(típus paraméternév,...);`

A delegált referencia típus. Ha paramétert is megadunk, akkor a paraméter nevét is meg kell adni.

Példa:

```
delegate int pelda(int x, string s);
```

Az előbbi példában tehát a *pelda* delegált típust definiáltuk. Ez olyan függvénytípus, amelyiknek egy egész és egy szöveg paramétere van, és eredményül egy egész számot ad.

Egy delegált objektumnak vagy egy osztály statikus függvényét, vagy egy osztály példányfüggvényét adjuk értékül. Delegált meghívásánál a hagyományos függvényhívási formát használjuk. Ezek után nézzünk egy konkrét példát:

Példa:

```
using System;
class proba
{
    public static int negyzet(int i)
    {
        return i*i;
    }
    public int dupla(int i)
    {
        return 2*i;
    }
}
```

```

class foprogram
{
    delegate int emel(int k); // az emel delegált definiálása

    public static void Main()
    {
        emel f=new emel(proba.negyzet);
                                // statikus függvény lesz a delegált
        Console.WriteLine(f(5)); // eredmény: 25
        proba p=new proba();
        emel g=new emel(p.dupla);
                                // normál függvény lesz a delegált
        Console.WriteLine(g(5)); // eredmény: 10
    }
}

```

A delegáltakat a környezet két csoportba sorolja. Ha a delegált visszatérési típusa *void*, akkor egy delegált több végrehajtandó függvényt tartalmazhat (*multicast*, összetett delegált), ha nem *void* a visszatérési típus, mint a fenti példában, akkor egy delegált csak egy végrehajtandó függvényt tud meghívni (*single cast*, egyszerű delegált).

Az egyszerű és összetett delegáltakra nézzük a következő példát:

Példa:

```

using System;
class proba
{
    public static int negyzet(int i)
    {
        return i*i;
    }
    public int dupla(int i)
    {
        return 2*i;
    }
    public int tripla(int i)
    {
        return 3*i;
    }
    public void egy(int i)
    {
        Console.WriteLine(i);
    }
    public void ketto(int i)
    {
        Console.WriteLine(2*i);
    }
}

```

VI. Függvények

```
class foprogram
{
    delegate int emel(int k);
    delegate void meghiv(int j);

    public static void Main()
    {
        emel f=new emel(proba.negyzet);
        Console.WriteLine(f(5));
        proba p=new proba();
        emel g=new emel(p.dupla);
        Console.WriteLine(g(5));
        emel h=new emel(p.tripla);
        g+=h;           //g single cast, g a tripla lesz
        Console.WriteLine(g(5));           // eredmény: 15
        meghiv m=new meghiv(p.egy);       // multicast delegált
        m+=new meghiv(p.ketto);
        m(5);           // delegált hívás, eredmény 5,10
    }
}
```

Ha függvénytípus (delegált) deklarálunk, akkor értékadás esetén a mutató típusa határozza meg, hogy a fordítónak melyik aktuális függvényt is kell az azonos nevék közül választania.

```
delegate double vmut(double,double);
// vmut olyan delegált amelyik egy valós értéket visszaadó,
// és két valós paramétert váró függvényt jelent

delegate int emut(int,int);
// emut olyan delegált, amelyik egy egész értéket visszaadó
// , és két egész paramétert váró függvényt jelent
vmut mut=new vmut(maximum); // valós hívás
Console.WriteLine(mut(3,5));
```

A *multicast* delegált (típus) definiálási lehetőséget, igazítva az igényeket az eseményvezérelt programozási környezethez (mind a Windows, mind az X11 ilyen), az egyes objektumokhoz, típushoz úgy kapcsolhatjuk, hogy esemény típusú mezőt adunk hozzájuk. Ezt az alábbi formában tehetjük meg:

public event meghiv esemeny;

ahol a *meghiv* összetett delegált. Az esemény objektum kezdő értéke *null* lesz, így az *automatikus meghívás (esemeny(...))* nem ajánlott!