



**BME** **KJT**  
Budapesti Műszaki és Gazdaságtudományi Egyetem  
Közlekedés- és Járműirányítási Tanszék

**Számítástechnika I.**

BMEKOKAA152

BMEKOKAA119

**Infokommunikáció I.**

BMEKOKAA606

**Dr. Bécsi Tamás**

**5. előadás**

# Tömbök tömbje

```
int[][] haromszog= new int[5][];  
for(int i = 0; i < haromszog.Length; i++)  
{  
    haromszog[i] = new int[i+1];  
    for(int j = 0; j < haromszog[i].Length; j++)  
        haromszog[i][j] = j;  
}
```

```
haromszog[0]: { 0 }  
haromszog[1]: { 0 , 1 }  
haromszog[2]: { 0 , 1 , 2 }  
haromszog[3]: { 0 , 1 , 2 , 3 }  
haromszog[4]: { 0 , 1 , 2 , 3 , 4 }
```

# Tömbök átméretezése

- `public static void Resize<T>( ref T[] array, int newSize )`
- Példa:  
`int[] a=new int[20];`  
`Array.Resize<int>(ref a, 22);`

# Struktúrák

Hozzáférési szintek:

- **private** csak struktúrán belülről érhető el
- **public** bárki elérheti ezt a mezőt
- **internal** programon belülről (assembly) elérhető

```
struct pont
{
    public int x;
    public int y;
};
```

# Struktúrák

- A struktúra elemeihez hozzáférni a `.` struktúratag operátorral tudunk.
- A struktúra tag operátor a legmagasabb precedenciájú, csak a `[]`, `()` zárja ki.
- A struktúrák esetében csak az értékadás operátort használhatjuk.
- A struktúrák érték típusok, értékadásnál másolás történik

```
Pont p, p1;  
p = new Pont();  
p.x = 1;  
p.y = 2;  
p1 = p;
```