

9 Tömbök (array of ...)

1 Tömbök (array of ...)

Eddigi programjainkban **egyedi változókkal** dolgoztunk, mindegyik változónak külön neve volt. Pl.

```
var a,b,c,i,n:integer;
```

Sok feladatot lehetetlen egyedi változókkal megoldanunk. Gondoljunk például egy névsor tárolására, melyben akár 100-200 név is lehet.

Ehhez szükségünk van olyan adattípusra, melyben több adatot tárolhatunk. Az eddig megismert típusok **elemi típusok** voltak. Ezekből az elemi típusokból adatszerkezeteket, **összetett típusokat** építhetünk. Ilyen összetett típus a tömb is.

A **tömb** típusának megadása:

```
var tömb_neve : array [ indexhatárok ] of alaptípus ;
```

Az *indexhatárok* megadják a tömb méretét (hogy hány elemet tárolhatunk benne), az *alaptípus* pedig az elemek típusát (ez lehet pl. **integer**, **string**, **char**, **boolean**, **byte**, stb).

Például:

```
var a:array [1..7] of string;
```

a[1]	a[2]	a[3]	a[4]	a[5]	a[6]	a[7]
Peter	Agnes	Ivan	Jozsef	Eva	Katalin	Timea

Ez létrehoz egy 7-elemű, **string**ekből álló tömböt. Ebben tárolhatunk például egy 7 elemből álló névsort. A tömb egyes elemeit az elemek indexén keresztül érhetjük el úgy, hogy a tömb neve után szögletes zárójelben megadjuk az elem **indexét**. A tömbünk tehát a következő elemekből áll: **a[1], a[2], ..., a[7]**.

A tömb beolvasása történhet egy egyszerű ciklussal:

```
for i:=1 to 7 do begin
  write('Add meg a(z) ',i, '. nevet: ');
  readln(a[i]);
end;
```

Feladat: Készíts egy olyan programot, amely 10 egész számot beolvas egy tömbbe, és kiírja azokat fordított sorrendben!

program szamtomb;

```
var szam : Array[1..10] of integer;
```

```
i :integer;
```

```
begin
```

```
writeln('Add meg a tíz számot!');
```

```
for i:=1 to 10 do
```

```
  begin
```

```
    writeln('A ',i, '.szám:');
```

```
    readln(szam[i]);
```

```
  end;
```

```
writeln('A számok az utolsótól az első számig:');
```

```
for i:=10 downto 1 do
```

```
  begin
```

```
    writeln('A(z) ',i:2, ',. szám', szam[i]:3);
```

```
  end;
```

```
readln;
```

```
end.
```