

---

# A C programozási nyelv IV. Deklaráció és definíció

Ficsor Lajos

Miskolci Egyetem

Általános Informatikai Tanszék

# Definíció és deklaráció

## Definíció:

- meghatározza valamely objektum típusát, méretét
- hatására helyfoglalás történik

## Deklaráció:

- valamely objektum típusát, méretét (alapvető jellemzőit) határozza meg
- hatására a deklarált azonosító használható hibajelzés nélkül
- a definíció egyben deklaráció is

# Alapszabály

- Minden azonosítót a teljes programot tekintve pontosan egyszer kell definiálni
  - kivétel: egy azonosító újradefiniálása (beágyazott hatáskörben)
- Csak deklarált azonosítók használhatók
  - kivétel: egy ( -el követett azonosítót a C nyelv egy **int**-et visszaadó függvény azonosítójának tekinti
  - **Ne használjuk ki ezt a szabályt!** (A függvényeket is mindig deklaráljuk!)

# A deklaráció formája

- `tárolási_osztály` típusnév  
`deklarátor_specifikátor`
- a `tárolási osztály` és a `típusnév` közül az egyik elmaradhat, ilyenkor a megfelelő alapértelmezés lép életbe.

# Típusnév

## Típusnév lehet:

- az alaptípusok ismertetésénél felsoroltak valamelyike
- struktúra- és unió definíció neve
- **typedef**- el definiált típusnevek

Alapértelmezés: **int**

# A deklarátor specifikátor 1.

az alábbiak valamelyike lehet:

Formája

**azonosító**

**azonosító [állandó kifejezés]**

**azonosító [ ]**

**azonosító ( )**

**a fentiek, előttük \*- al**

**( \* azonosító ) ( )**

Jelentése

alaptípus

tömb

tömb

függvény

fentiek

mutatói

függvény-

mutató

# A deklarátor specifikátor 2.

## Korlátozások:

- Tömb csak az alábbiakból képezhető:
  - alaptípusok
  - mutatók
  - struktúrák
  - uniók
  - tömbök
- Függvényérték nem lehet:
  - tömb
  - unió
  - függvényde lehet a fentiek bármelyikét megcímző mutató!

# A deklarátor specifikátor 3.

Példák:

<code>int t[ ]</code>	egészeket tartalmazó tömb
<code>int *t[ ]</code>	egészeket megcímző mutatókat tartalmazó tömb
<code>int f( )</code>	egészt visszaadó függvény
<code>int *f( )</code>	egészt megcímző mutatót visszaadó függvény
<code>int *( f( ) )</code>	egészt visszaadó függvényt megcímző mutató
<code>int ( *f( ) ) [ ]</code>	olyan tömb, amelyeknek elemei fenti típusú függvény-mutatók



# Tárolási osztályok

- A definiált objektum érvényességi körét és élettartamát (vagy tárolási módját) határozza meg.

**auto** Lokális változó egy blokkra nézve. Értéke a blokkba való belépéskor határozatlan, a kilépéskor megszűnik.

**regiszter** Olyan **auto** változó, amelyet gyakran kívánunk használni. Nem alkalmazható rájuk az **&** operátor

**extern** Általános érvényű változó, a program különböző részeiben is érvényes.

**static** Értéke megmarad. (Lehet belső vagy külső.)  
Érvényességi köre korlátozott.

# Külső azonosítók

- **Külső definíció** az, amely minden függvényen kívül helyezkedik el. A függvénydefiníció mindig külső. A külső definíció több függvény számára is ismert.
  - **extern** tárolási osztály (**alapértelmezett**): a teljes programra érvényes (globális)
  - **static** tárolási osztály: csak a definíciót tartalmazó forrásfile-ra érvényes
- Adatok külső definíciói kerülendők!

# Belső azonosítók

- Egy blokk elején helyezkedik el.
  - **auto** tárolási osztály (**alapértelmezett**)
  - **register** tárolási osztály: gyakran használni kívánt érték, gyors elérésre optimalizálándó
  - **static** tárolási osztály: a blokkba való belépéskor az előző értéke megmarad, de a blokkon kívül nem használható.
- Ha egy befoglaló blokkban ugyanazt az azonosítót deklaráljuk, a blokkon kívüli deklaráció felfüggesztődik. (*Újradefiníálás*)

# Inicializálás (kezdőérték adás) 1.

- Inicializálás nélkül:
  - a külső és a statikus változók értéke garantáltan 0
  - az automatikus és regiszterváltozók értéke határozatlan.
- Egyszerű változó inicializálása:  
`tár_o típus azonosító = kif`

# Inicializálás (kezdőérték adás) 1.

- Összetett objektum inicializálása:  
 $\text{deklaráció} = \{ \text{kif}, \text{kif}, \dots, \}$ 
  - A kifejezések az elemek sorrendjében adódnak át.
  - Ha számuk kevesebb, mint amit a deklaráció meghatároz a maradék elemek 0-val töltődnek fel.
  - Ha több, hibajelzést kapunk.
  - Tömbök esetén a deklarációból az első index felső határa elmaradhat, ezt ekkor a kifejezések száma határozza meg.