

---

# **A C programozási nyelv II. Utasítások. A függvény.**

Ficsor Lajos

Miskolci Egyetem

Általános Informatikai Tanszék

# Kifejezés utasítás, blokk

Kifejezés utasítás

**kifejezés;**

Összetett utasítás

{  
**utasítások**  
}

Blokk

{  
**deklarációk**  
**utasítások**  
}

# Feltételes utasítás

**if** ( kifejezés ) utasítás;

vagy

**if** ( kifejezés )

utasítás1;

**else**

utasítás2;

# Ciklusutasítások 1.

Elöltesztelő, iteratív

**while** (kifejezés) utasítás;

Hátultesztelő, iteratív

**do** utasítás **while** (kifejezés);

# Ciklusutasítások 2.

Általános

```
for ( k1; k2; k3 ) utasítás;
```

jelentése:

```
k1;
```

```
while (k2)
```

```
{
```

```
utasítás;
```

```
k3;
```

```
}
```

# Többirányú elágazás 1.

```
switch (kif)
{
  case (ak1):
    u1;
  case (ak2):
    u2;
  case (akn):
    un;
  default:
    un+1;
};
```

<-- ez nem kötelező!

# Többirányú elágazás 2. (Működés)

- kiértékelődik a `kif`
- a program azon első `case` szerkezet utáni utasítással folytatódik, amelyben szereplő állandó kifejezés értéke egyenlő `kif` értékével.
- ha a fenti feltétel egy esetben sem teljesül, a `default` utáni utasítással folytatódik (ha van ilyen címke).
- minden egyéb esetben a `switch` utasítást követő utasítással folytatódik a program.

# Vezérlésátadás

## **break ;**

- Befejeződik a break-et körülvevő legbelső `while`, `for`, `do` vagy `switch` utasítás végrehajtása, és a vezérlés átadódik a következő utasításra.

## **continue ;**

- A vezérlés a körülvevő legbelső `while`, `do` vagy `for` utasítás ciklusvégeére adódik át.



# Visszatérés függvényből

- **return ;**
- A függvény végrehajtása befejeződik, és a hívó függvényhez tér vissza a vezérlés. A függvény értéke definiálatlan.
- **return kifejezés ;**
- Visszatérés a hívó függvényhez, a függvény értéke a kifejezés lesz. (Típuskonverzióval, ha szükséges.)

# A függvény fogalma

- A függvény (mint minden programozási nyelvben) utasítások egy névvel azonosítható csoportja, amely megadott paramétereken képes műveleteket végezni.
- A tipikus C nyelvű program sok, viszonylag egyszerű függvény összessége.

# Függvény definíció 1.

Formája:

```
típus  név(formális paraméterlista)
{
    lokális deklarációk
    utasítások
}
```

## Függvény definíció 2.

- A formális paraméterlista
  - `típus azonosító`

vagy

- `típus tömbnév[ ]`

párok, vesszővel elválasztva.

(A fenti felsorolás nem teljes!)

Ha nincs paraméter, a paraméterlista helyére a `void` alapszó írandó.

# Függvény definíció 3.

- A visszatérési érték típusa lehet:
  - típusnév
  - típusnév\* (pointer)
  - struktúranév és struktúra pointer
  - függvény pointer
  - unió pointer
  - **void** (nincs visszatérési érték)
- A visszatérési érték nem lehet tömb .

# Függvény deklaráció (prototípus)

- A függvényt a használata előtt deklarálni kell.

Formája:

```
típus név(formális paraméterlista);
```

## Megjegyzés

A C nyelv ettől enyhébb szabályokat ír elő, de a helyes programozási stílus elsajátítása érdekében fogadjuk el ezt a szigorúbb szabályozást.

# Függvény hívása

**név (aktuális paraméterlista)**

- A függvényhívás állhat
  - pontosvesszővel lezárva, (ekkor a visszaadott érték - ha volt - elvész)
  - kifejezés részeként.
- Az aktuális paraméterlista kifejezések vesszővel elválasztott listája.
- A zárójelpár kiírása akkor is kötelező, ha nincs paraméterlista!

# Paraméterátadási mechanizmus 1.

**Csak érték szerinti paraméterátadás!** Folyamata:

- kiértékelődik az aktuális paraméter kifejezés
- a kifejezés értéke a formális paraméter típusára konvertálódik a szokásos típuskonverzió szabályai szerint
- a formális paraméter megkapja kezdőértéknek ezt az értéket
- végrehajtnak a függvény törzsében felsorolt utasítások



# Paraméterátadási mechanizmus 2.

## Következmények:

- A formális paraméterek a függvényre nézve *lokális* változók, amelyek az aktuális paraméter kifejezés értékével inicializálódnak.
- A formális paraméterek a függvényen belül kaphatnak más értéket is, de ennek az aktuális paraméterre semmi hatása nincs.
- Az aktuális paraméterek tetszőleges (de konvertálható típusú) kifejezések lehetnek.