

# INFORMATIKA LABORATÓRIUM I.

Alprogramok

# Függvények

- A C nyelvben a programot függvények alkotják.
- A függvény a szubrutinhoz hasonló, csak nem annyira általános fogalom.
- Akár a szubrutinnak, a függvénynek is lehet tetszőleges számú bemenő értéke, csak míg a szubrutin tetszőleges számú visszatérési értéket szolgáltat, a függvény csak egyet.
- A függvény maga is hívhat függvényeket, sőt önmagát is meghívhatja közvetlenül, vagy közvetve (rekurzió!)
- Kitüntetett függvény a `main()`: ez a program belépési pontja.
- Nincs akadálya annak, hogy bármely függvény újra meghívja a `main ()`- t.
- A C nyelvben a függvények hívása egyszerű: le kell írni a nevüket. Az általuk szolgáltatott értéket értékadással vehetjük át.  
pl.: `c=getchar()` ;

# Függvények

- ◉ A paraméterek átadása zárójeleken belül történik. Ha nincs paraméter, akkor is kell a zárójel.
- ◉ Ha a zárójeleket elhagyjuk, akkor a változónevek között keresi, nem pedig a függvénynevek között a fordító.

A függvény definiálásának alapvető formája:

```
típus függvény-név (paraméter lista)
paraméter deklarációk
{
    a függvény törzse
}
```

- ◉ A típus írja elő annak az értéknek a típusát, amelyet a függvényből való visszatéréskor a **return** utasítás szolgáltat.
- ◉ Az érték bármilyen típusú lehet. Ha nincs típus, akkor a függvény egy egész értékkel tér vissza.
- ◉ Ha a függvény nem ad vissza értéket akkor **void** típusúnak kell deklarálni.

# Paraméterek és argumentumok

- ⦿ A hívó szempontjából az átadott értékek **paraméterek**.
- ⦿ A hívott szempontjából a kapott értékek **argumentumok**.
- ⦿ Amikor a függvénybe belépünk, akkor válnak a paraméterek argumentumokká.
- ⦿ A paraméterek és argumentumok számban és típusban meg kell, hogy egyezzenek.
- ⦿ A hívó által létrehozott paraméterek és a hívott által való értelmezés teljesen független egymástól. Ha tehát számban, vagy típusban eltérő paramétereket adunk meg, azt senki sem veszi észre.

# A függvény típusa és a return utasítás

- Egy függvény adhat vissza értéket a hívó számára. A visszaadott érték típusa a függvény típusával egyezik. Érték visszaadására a **return** utasítás szolgál.
- Ha egy függvénynek nem adjuk meg a típusát, akkor az automatikusan int típusú lesz.
- Azt a függvényt, amely nem ad vissza értéket, **void** típusúnak kell definiálnunk.

# Érték és hivatkozás szerinti hívás

A függvények alapvetően kétféleképpen tudják átadni az argumentumokat, azaz kétfajta hívást különböztetünk meg:

- **Érték szerinti hívás:**  
Az argumentum értéke átmásolódik a formális paraméterbe, de az argumentum értéke nem változik meg.
- **Hivatkozás szerinti hívás:**  
Az argumentum címe másolódik át a formális paraméterbe. A függvényen belül a címet használjuk fel aktuális argumentumként, így annak tartalma megváltozhat.

## Return utasítás

Két fontos szerepe van.

- Függvényből való kilépés úgy, hogy a vezérlés a hívás helyére adódjon vissza.
- Adhat egy értéket vissza.

# Példa

Érték szerint átadott paraméterek:

```
int novel(int szam)
{
    return szam + 1;
}
```

Globális paraméterek

```
int szam = 0;

void novel()
{
    szam = szam + 1;
}
```

Cím szerint átadott paraméterek

```
void novel(int *szam)
{
    *szam = *szam + 1;
}

main()
{
    int szam = 0;
    novel(&szam);
}
```

# Gyakorló feladat

- Készítsünk olyan alprogramot, amely kiszámolja egy háromszög területét az oldalak ismeretében!



# Gyakorló feladat

- Készítsünk olyan alprogramot, amely kiszámolja egy másodfokú egyenlet megoldását. A függvénynek legyen visszatérési értéke a valós megoldások száma,  $x_1$ ,  $x_2$  megoldást paraméterekkel oldjuk meg!

# Gyakorló feladat

Készítsünk olyan alkalmazást, ami alprogramok segítségével kezeli egy számokat tartalmazó tömböt. Oldjuk meg *alprogramokkal* a következő funkciókat

- Töltsük fel a tömböt többféle módon: „kézi” feltöltés, „default” értékkel, véletlenszám-generátorral.
- Írassuk ki az adatokat!
- Határozzuk meg a tömb összegét, maximumát, minimumát, átlagát
- Készítsünk alprogramot, amely eldönti, hogy az adott elem szerepel-e a tömbben
- Készítsünk alprogramot, amely stringgé alakítja a tömb értékeit egy megadott szeparátorral elválasztva!

# Gyakorló feladat

Budapesti színházak adatait tartalmazza a következő táblázat:

	Erkély	Páholy	Földszint elől	Földszint hátul
Madách	60	20	180	180
Víg	80	40	220	200
Katona József	0	12	110	100
Józsefvárosi	0	10	80	100
Opera	120	35	210	200
Erkel	250	40	300	300

- Írassuk ki az adatokat táblázatosan! (alprogram neve: kiir)
- Készítsünk telhazbevetel nevű függvényt, amely a paraméterként megkapott színház és jegyár ismeretében visszaadja a teltházás bevétel összegét!
- Írassuk ki, hogy hol a legnagyobb a páholy a földszint! (függvény neve: maxPaholy)
- Írassuk ki, hogy hány hely van földszinten elől! (függvény neve: fszElolDarab)