



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

# Programozás C- és Matlab nyelven

## C programozás kurzus

### BMEKOKAM603

### Struktúrák

Dr. Bécsi Tamás

5. Előadás

# 6. Struktúrák

- A struktúra egy vagy több, esetleg különböző típusú változó együttese, amelyet a kényelmes kezelhetőség céljából önálló névvel látunk el.
  - (Néhány nyelvben az így értelmezett struktúrát rekordnak nevezik.)
- A struktúra bevezetése segíti az összetett adathalmazok szervezését, ami különösen nagy programok esetén előnyös, mivel lehetővé teszi, hogy az egymással kapcsolatban lévő változók egy csoportját egyetlen egységként kezeljük, szemben az egyedi adatkezeléssel.

# 6.1. Alapfogalmak

- A struktúra deklarációját a **struct** kulcsszó vezeti be, amelyet kapcsos zárójelek között a deklarációk listája követ.  
A struct kulcsszót opcionálisan egy név, az ún. **struktúracímke** követheti. Ez a címke vagy név azonosítja a struktúrát és a későbbiekben egy rövidítésként használható a kapcsos zárójelek közötti deklarációs lista helyett.

# 6.1. Alapfogalmak

## Példa: címkézett struktúra

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

- A kifejezés egy típust definiál, ahol a pont a struktúracímke, x és y a tagok. Ha a struktúra címkézett volt, akkor a címke a későbbi definíciókban a struktúra konkrét előfordulása helyett használható:

```
struct pont pt;
```

- Amely egy pont típusú változót definiál, amely akár közvetlenül inicializálható a tagok értékeinek felsorolásával:

```
struct pont pt={ 320 , 200 };
```

# 6.1. Alapfogalmak

- Egy kifejezésben az adott struktúra egy tagjára úgy hivatkozhatunk, hogy

`struktúra-név.tag`

- A pont struktúratag operátor összekapcsolja a struktúra és a tag nevét. A pt pont koordinátáit pl. úgy írathatjuk ki, hogy

```
printf("%d, %d", pt.x, pt.y);
```

# 6.1. Alapfogalmak

- A struktúrában felsorolt változóneveket a struktúra *tagjainak* nevezzük. Egy struktúra **címkéje** (neve), ill. egy tagjának a neve és egy közöséges (tehát nem struktúratag) változó neve lehet azonos, mivel a programkörnyezet alapján egyértelműen megkülönböztethetők:

```
struct str{int str};
```

```
struct str str;
```

- Ez szintaktikailag helyes, ám nem ajánlott.
- Ugyanaz a tagnév előfordulhat különböző struktúrákban

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

```
struct kor{int x;int y;int r};
```

# 6.1. Alapfogalmak

## Példa: nem címkézett struktúra

- Egy struct deklaráció egy típust is definiál. A jobb oldali, záró kapcsos zárójel után következhet a változók listája, hasonlóan az alapadattípusok megadásához. Így pl. a  
`struct { ... } x, y, z;`
- Közvetlenül hoz létre x,y,z struktúra változókat, és foglal le helyet számukra a tárterületen.

# 6.1. Alapfogalmak

- A struktúrák egymásba ágyazhatók. Például egy téglalap az átlója két végén lévő pontpárral írható le:

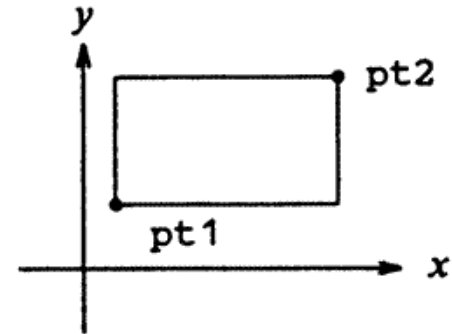
```
struct tegla {  
    struct pont pt1;  
    struct pont pt2; };
```

- Ennek alapján a tegla struktúra két pont struktúrából áll. Ha az abra struktúrát úgy deklaráljuk, hogy

```
struct tegla abra;
```

- akkor az  
`abra.pt1.x`

- hivatkozás az abra pt1 tagjának x koordinátáját jelenti.





# Köszönöm a figyelmet

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

