

### 3 előadás

`int Fv (int a, int b)` } c++-ban  
`int Fv (float a, float b)` } lehetséges

A fordító a lebegőpontos típusokat alapból double-ként kezeli.

Globális változó alapértéke 0 (ha nem definiáljuk).

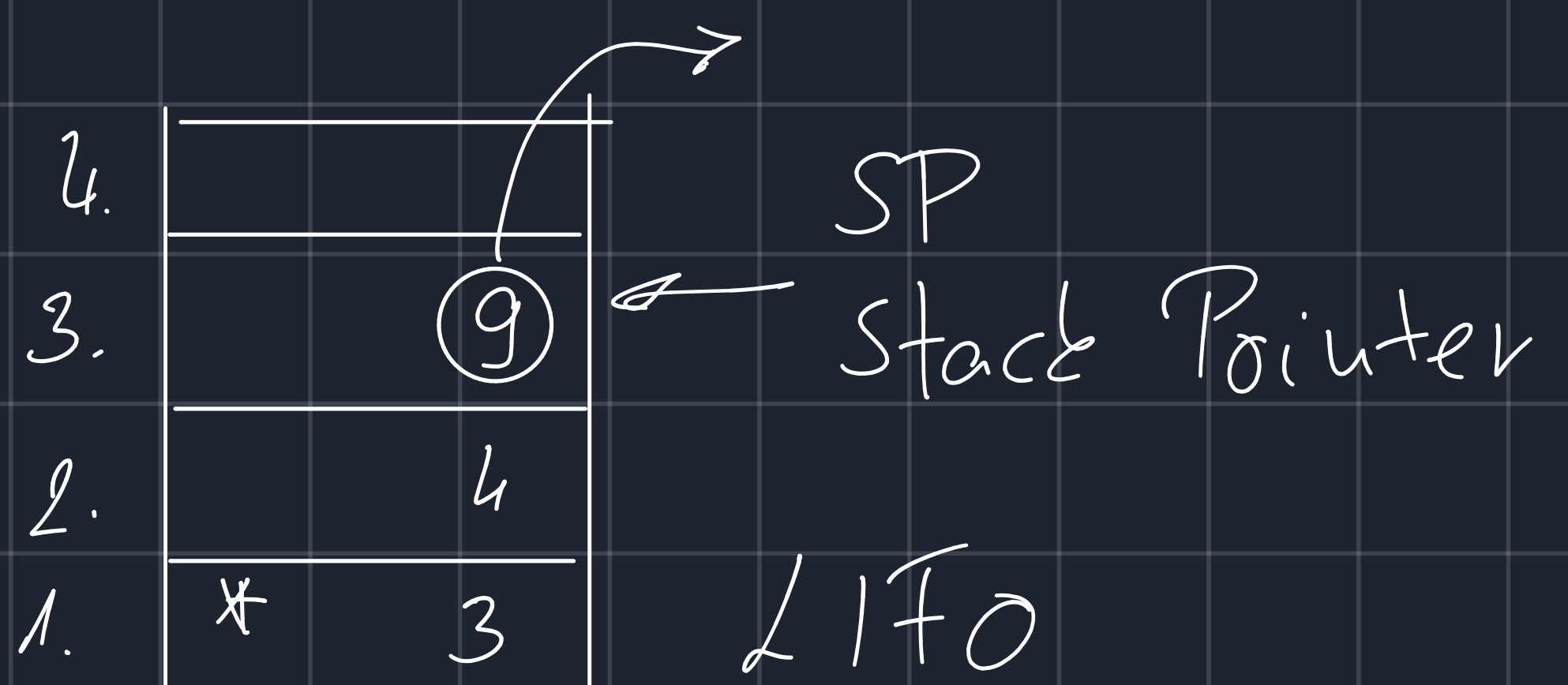
Lokális változó elfedi a globális változó értékét.

```
int a;  
  
int main() {  
    int a = 3, // 3  
    ret 0;  
}
```

:: 0 → globális  
hivatkozási  
operátor

```
int main() {  
    _____  
    _____  
    _____  
    * cout << F( , );  
}
```

```
int F() {  
    ** ↓()i  
}
```



Hívási level

Hatékonyság  $\rightarrow$  ctime

```
clock - t kezd, veg;  
kezd = clock();
```

```
veg = clock();
```

```
cout << (float) (veg - kezd) / CLK_TCK << endl;
```

Inline függvények  $\rightarrow$  "sorok között"

jóval gyorsabb  $\leftarrow$  a fordító a hívás helyén  
lesz befűrdítja a fő függvény  
soraiiba.

Macrók használata (ő kell paraméterezni)

```
#define osszeszoroz(x,y) x*y
```

Nagyon fontos  
zárojelezni.

$$3 + 3 * 8 - 2 \Rightarrow 25$$

Fájlkezelés  $\rightarrow$  szöveges fájlkezelés  
 $\rightarrow$  bináris fájlkezelés



Gyaz.txt

open()  
close()  
fail()

<fstream>

ofstream ki("gyaz.txt")

("C:\\gyaz.txt")

string  
feloldas  
miatt kell  
a második \-jel.

```
ifstream be("misi.txt");
```

```
ki << "misi" << endl;  
ki.put('b'); // egy betűt tesz ki.
```

```
ifstream be("miska.txt");  
be.get() // egy betűt olvas be  
be >> szo; // beemelő operátor  
getline(be, szo);
```

```
// be.getline(nev, 80);
```



```
char nev[80];
```