

# Указатели

гл. ас. д-р. Нора Ангелова

---

# Указатели

- Оператор &

&<променлива>

<променлива> ::= вече дефинирана променлива.

Семантика

*Намира адреса на променливата.*

# Указатели

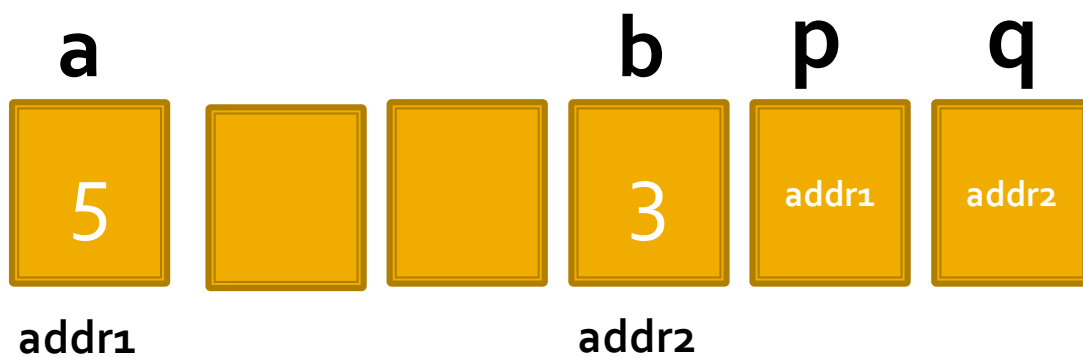
$T^* \text{ <променлива> [= <стойност> ]_{опц}}$

$T^*$  – указател към променлива от тип  $T$ .

- $T$  – име или дефиниция на тип.
- $\text{<променлива> ::=}$  идентификатор.
- $\text{<стойност>}$  – шестнадесетично число, представляващо адрес на данна от тип  $T$  или `nullptr` (NULL).

# Указатели

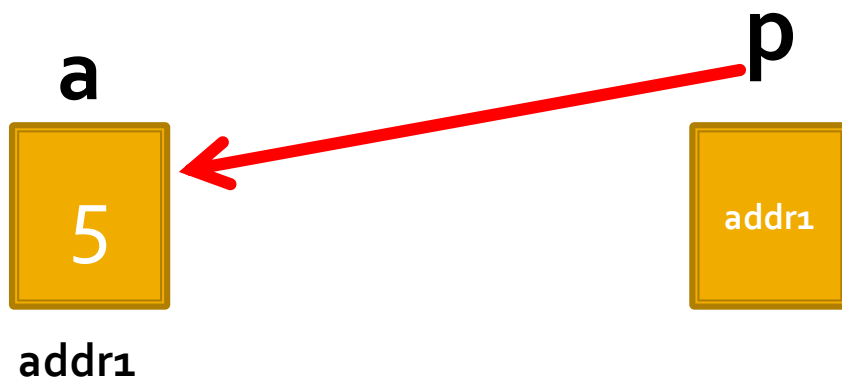
```
int a = 5;  
int b = 3;  
int *p = &a;  
int *q = &b;
```



# Указатели

```
int a = 5;  
int *p = &a;
```

```
      ТИП  
a (int) == 5  
&a (int *) == addr1
```

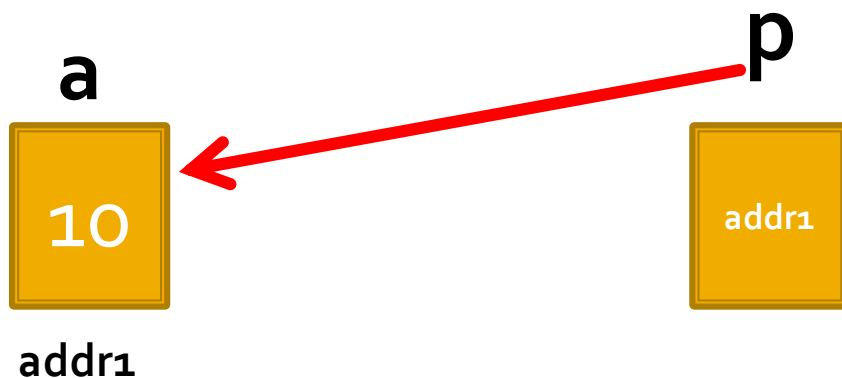


```
      ТИП  
p (int *) == addr1  
*p (int) == 5
```

# Указатели

```
int a = 5;  
int *p = &a;
```

```
*p = 10; // Присвоява нова стойност на a
```



# Указатели

Забележка

Дефиницията:  $T^* a, b;$

е еквивалентна на:

$T^* a;$

$T b;$

Дефиницията:  $T^* a, b;$

е еквивалентна на:

$T^* a;$

$T^* b;$

# Аритметични и логически операции

- +, -, ++, --, ==, !=, >, >=, <, <=

*\* Не е възможно въвеждане на данни от тип указател чрез оператора >>.*



# Адресна аритметика

```
int *p;  
double *q;
```

- `p = p + 1;` // премества указателя

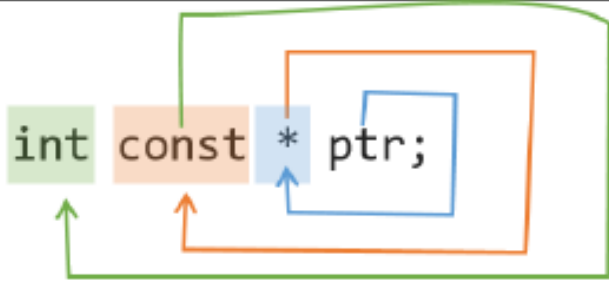
`p = p + 1*4,`

4 е броят на байтовете, необходими за записване на данна от тип `int`.

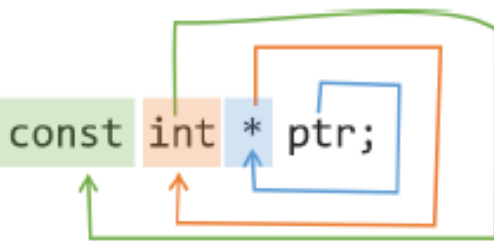
- `q = q + 1;` // премества указателя

`q = q + 1*8,`

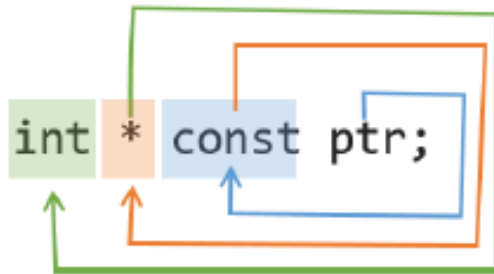
8 е броят на байтовете, необходими за записване на данна от тип `double`.



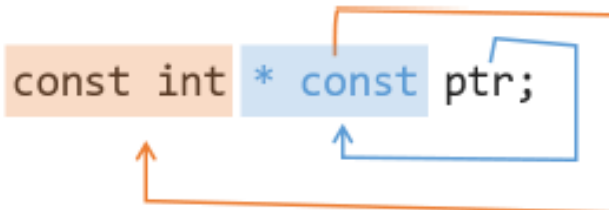
ptr is a **pointer** to **const int**



ptr is a **pointer** to **int constant** (i.e. const int)



ptr is a **const pointer** to **int**



ptr is a **constant pointer** to **const int**

---

Следва продължение . . .