

PUNTOR 1

T
int

*T
*int

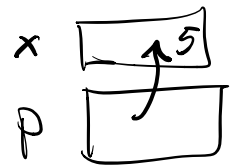
var
var

x
p

int
*int

p = &x

*p = 5



new (T)

↓
restitute *T

<u>var</u>	x	int
<u>var</u>	p	*int
<u>var</u>	q	**int

x = 7

p = &x

q = &p

p = new (int)

*p = 50

q = new (*int)

*q = &x

**q = 5

*q = new (int)

**q = 12

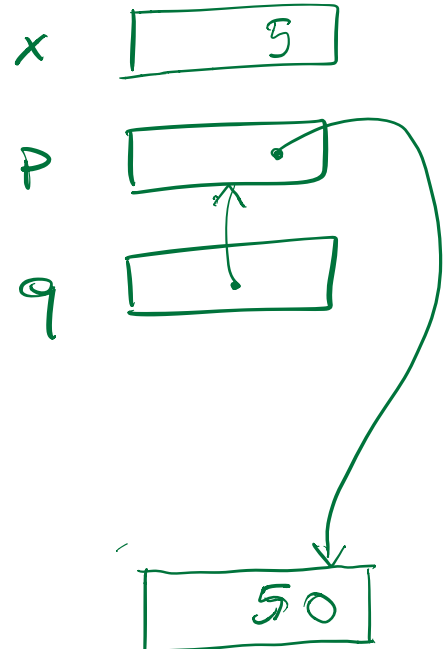
q = &p

func. Println(x, *p, **q)

5 50 50

x = *p

*p = x



TYPE

TYPE DEFINITION

type {id} {tipo}

non sono
sinonimi
→ conversioni

TYPE ALIAS

type {id} = {tipo}

sinonimo

type punt = *int
var x, y, z punt

type
var
var

x = 5

y = x

intero = int
 x intero
 y int

type
var
var

y = int (x)
 x = intero (y)

intero int
 x intero
 y int

```
func    main() {  
    var x int  
    x = 5  
    foo(&x)  
}
```

```
func    foo(p *int) {  
    . . .  
}
```

STRUCT

```

struct {
    id, id .... T
    id, id .... T
    ...
}
    
```

① var data Di Nascita struct {
 g, m, a int
 }

② type data struct {
 g, m, a int
 }
var data Di Nascita data

data Di Nascita

g	m	a
0	0	0

data DiNascita.g = 29

data DiNascita.m = 11

data DiNascita.a = 1968

data DiNascita.g, data DiNascita.m,
data DiNascita.a = 29, 11, 1968

```
type persona struct {  
    nome, cognome      string  
    altezza            int  
    peso               int  
    data DiNascita     data  
    ewo plo binario    float64  
}
```

var x, y persona

x

nome	cognome	altezza	peso	data DiNascita			ewo plo binario
				g	m	a	

$x.\text{nome} = \text{"Roberto"}$
 $x.\text{cognome} = \text{"Baldi"}$
 $x.\text{altezza} = 172$
 $x.\text{peso} = 72$
 $x.\text{data_di_Nascita}.g = 29$
 $x.\text{data_di_Nascita}.m = 11$
 $x.\text{data_di_Nascita}.a = 1968$

...

LETTURE DI TIPO STRUTTURA

① $\text{Var } d \text{ data}$
 $d = \text{data} \{ 29, 11, 1968 \}$
 $\text{Var } p \text{ persona}$
 $p = \text{persona} \{ \text{"Roberto"}, \text{"Baldi"}, 172, 72, \text{data} \{ 11, 5, 1967 \}, 61.5 \}$

② $d = \text{data} \{g: 29, m: 11\}$
 $p = \text{persona} \{nome: "Polo"\}$

DOVE SI DI CHIARANO
 I TIPI

type ---
func $f(\dots) \{$

var ---

⋮

}

ESERCIZIO

- Dichiarare un tipo data
- Scrivere una funzione
epoch
che data una data, restituisce
i giorni dall'epoca (1/1/1970)
- Scrivere un'altra funzione
che data due date calcola
i giorni
la prima e la seconda
intercorsi fra

Funzione che dato un anno restituisce una data che è l'ultimo giorno di febbraio di quell'anno.

ESERCIZIO

- Scrivere una funzione che prenda tre argomenti
 - un puntatore a una data
 - una stringa s
 - un intero x

La funzione deve modificare uno dei tre campi della data (il giorno se $s == "g"$, il mese se $s == "m"$, l'anno se $s == "a"$) ponendolo uguale a x .