

parameter formula

```

func increment (x int, k int) int {
    return x + k
}

```

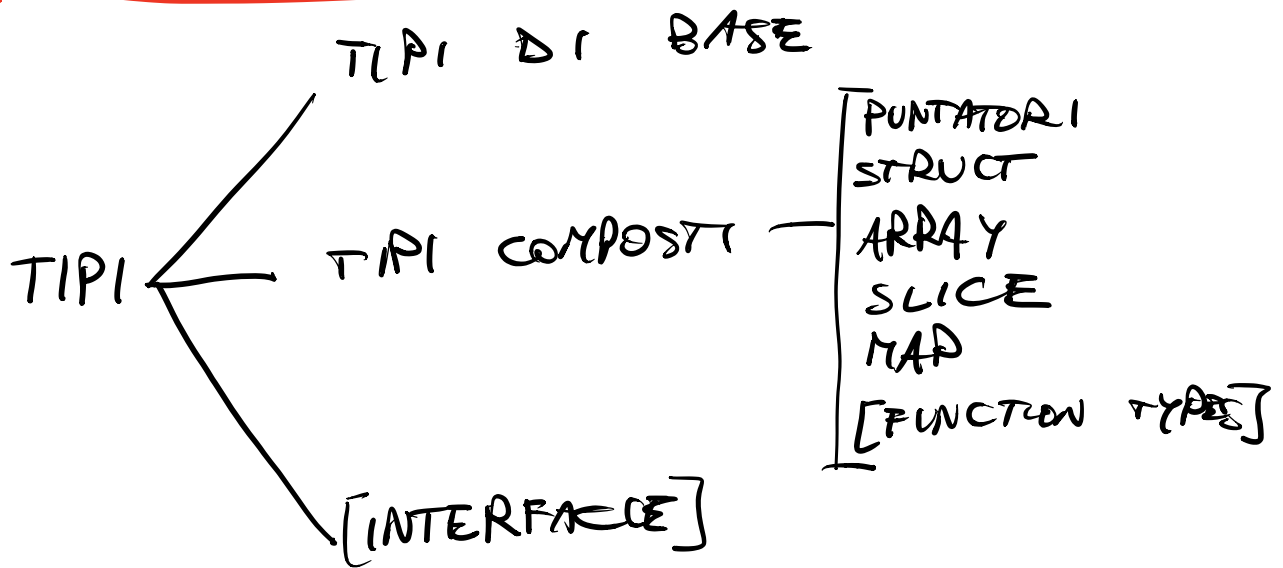
```

func main () {
    var y int
    y = 7
    y = increment(y, 15)
    fmt.Println(y)
}

```

CALL - BY - VALUE

TIP1 COMPOSTI



PUNTATORI

≡ locatione di memoria in cui si trova un valore

SINTASSI

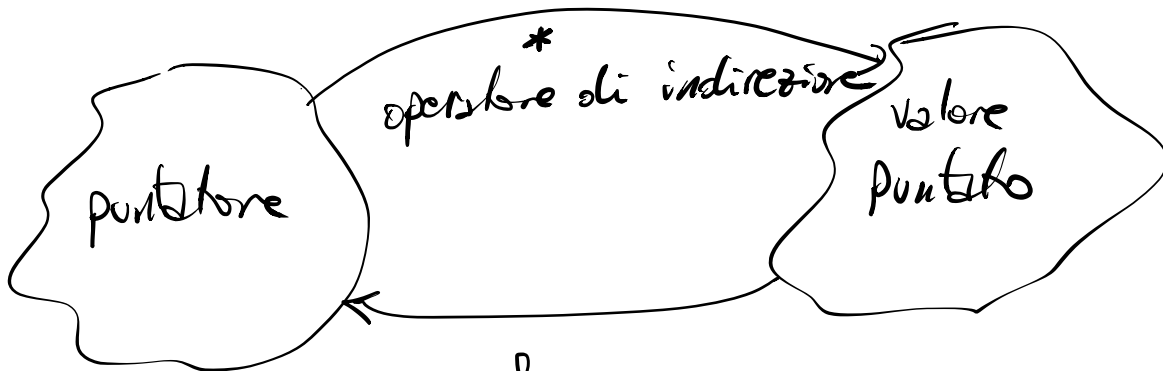
T è un tipo

* T

← puntatore a T
≡ rappresenta una locatione di memoria dove si trova un T

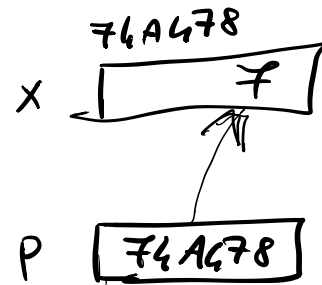
* int64
* string
* bool

var p *int64



&
operatore di indirizzare

var p *int
var x int
p = &x
*p = 7



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

x = 15

p = &x

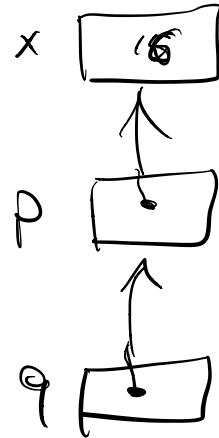
q = &p

**q ++

↑
Address of x

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

fun. Println(x, p, q)



A SCRIVETE UNA FUNZIONE
CHE DATI x E y
RESTITUISCE DIVISIONE
INTERA x/y E
RESTO (COME DUE
VARI RESTITUITI)

B STESSA COSA MA
I DUE VARI VENGONO
SCRITTI IN DUE
VARIABLE PASSE
COME PUNTORI