

# SLICE CHEATSHEET

DICHIARAZIONE

var x, y [int]

POPOLAZIONE (RIEMPIMENTO)  
DELLA SLICE

①  $x = \text{make}(\text{int}, n)$

②  $x = \text{append}(x, 7)$

$x = \text{append}(x, 13)$

$x = \text{append}(x, 2, 3)$



for  $i, t := \text{range } x \}$

$x[i] \leftarrow \text{lettura / scrittura}$   
 $t \leftarrow \text{lettura}$

}

## ALIAS

Due slice che condividono  
in tutto o in parte  
l'array sottostante.

var  $x, y$  [int  
 $x = \text{make}([int, 10])$

$y = x$  ← ALIAS

EVIARE DI AVERE IN GIRO  
ALIAS SE LA SLICE  
DEVE ANCORA ESSERE  
MODIFICATA (append !!!)

# ESEMPIO 1

- Scrivere un programma che data una slice di stringhe restituisce la stringa ottenuta concatenandone le iniziali.

Gianni | cade | dalla | finestra

Gcdf

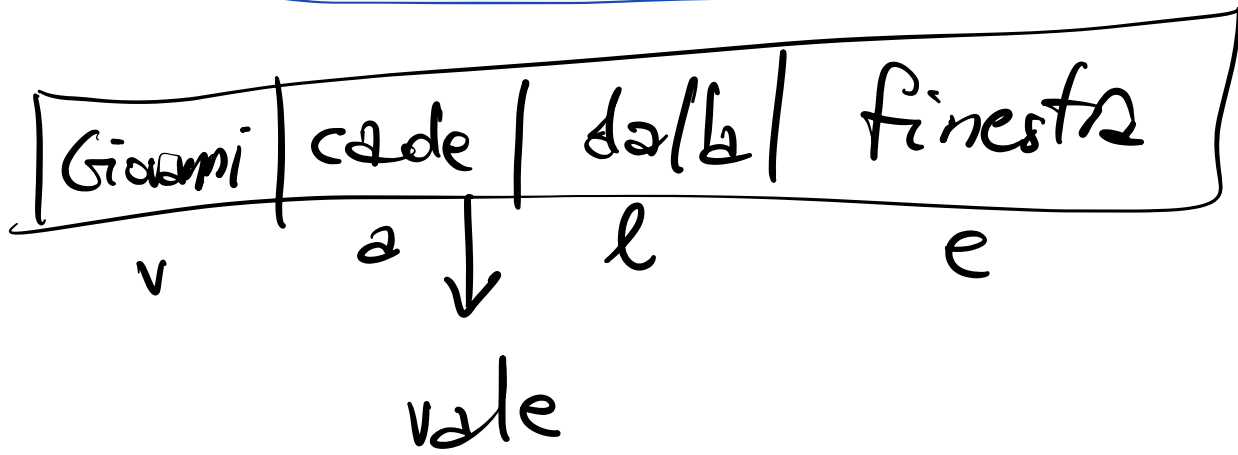
func acronimo (s []string) string {  
var t string

for \_, x := range s {  
t += string(x[0])  
}

return t

}

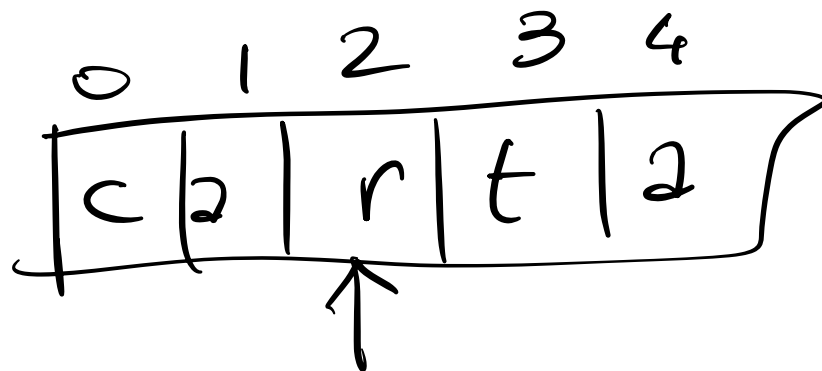
for i := 0; i < len(s); i++ {  
t += string(s[i][0])  
}



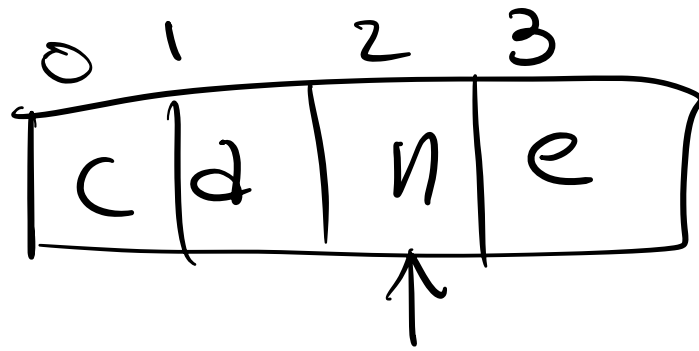
```

func veronimo (s []string) string {
    var t string
    for _, x := range s {
        n := len(x)
        t += string(x[n/2])
    }
    return t
}

```



$$5 \\ 5/2 = 2$$



$$4$$

$$4/2 = 2$$

```

func main () {
  var x []string
  x = []string { "Giovanni",
                 "cde", "dalla", "Finestra" }

  t := acronimo (x)
  fmt.Println (t)
}

```

OPPURE

```

func      main () {
    var      x [] string
    scanner := bufio.NewScanner
                (os.Stdin)
    scanner.Split (bufio.ScanWords)
    for      scanner.Scan () {
        x = append (x, scanner.Text())
    }
    t := acronym (x)
    fmt.Println (t)
}

```

---

```

$> ./pippo
Garibaldi e Vittorio
Eusquele
^D
GeVE
$>

```

# ARGOMENTI DA RIGA DI COMANDO

\$> go build pippo.go

\$> ./pippo



\$> ./pippo ciao ciao tre  
cavalli

\$> ./pippo Giovanni e Carlo sono  
morti

Go Csm

\$>

ARGOMENTI

os.Args

[]string



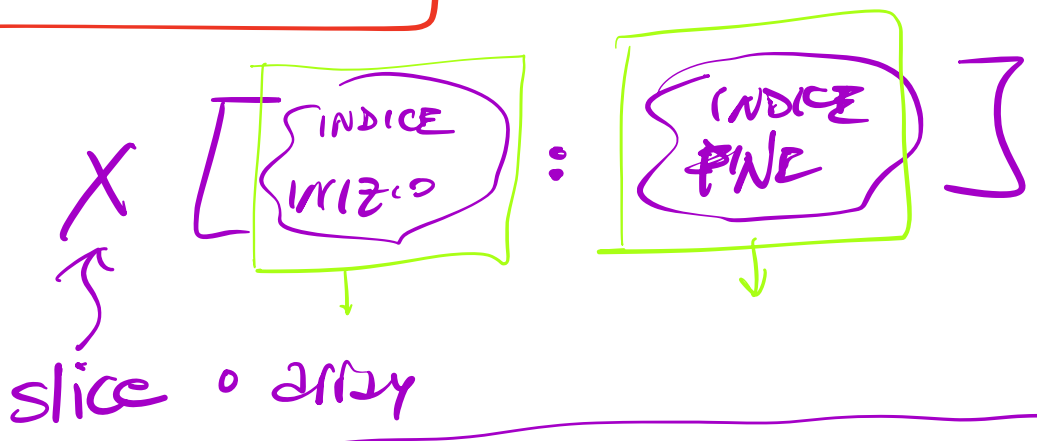
0) il nome del programma

1) } ARGOMENTI  
:  
:

func main () { <sup>SUBSLICING</sup>  
t := cronio (os.Args [1:])  
fmt.Println(t)  
}

↑  
NO

# SUBSLICING



sublice ↓ (alias parziale di x)  
 dell'indice  $\rightarrow$  **inizio**  
 compreso all'indice di  
**fine** escluso

	0	1	2	3	4
x	71	123	43	12	131

	0
x[3:4]	12

	0	1	2
x[1:4]	123	43	12

	0	1
x[:2]	71	123

	0	1
x[3:]	12	131

x [0:]

0	1	2	3	4
71	123	43	12	131

y := x[0:4]

y[0] = -1

---

```

func avg(x [int] float64) {
  s := 0
  for _, t := range x {
    s += t
  }
  return float64(s) / float64(len(x))
}

```

```

func main() {
  var a [int] {71, 123, 43, 12, 131}
  Println(avg(a))
}

```

func.  $\text{avg}(a)$   
 $\text{avg}(a[1:3])$   
 $\text{avg}(a[2:4])$

var  $b$  [3]int  
 $b = [\dots]$ int {10, 20, 30}

~~$\text{avg}(b)$~~   
 $\text{avg}(b[:])$

./date	5	1988	2010
--------	---	------	------

3/10/1989  
4/4/2003  
12/11/2001

⋮