var a, b [100]int

var x, y []int

x = make([]int, 100)

x[3] = 5

x = append(x, 7)

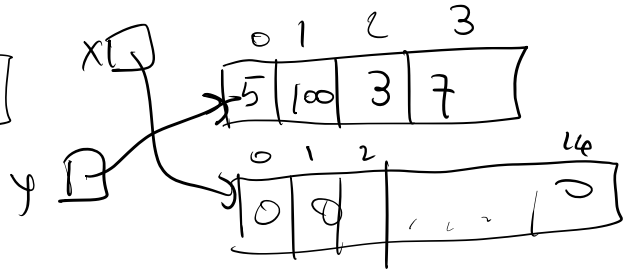
x = append(x, 15)

x = append(x, 100)

x = append(x, 3, 7)

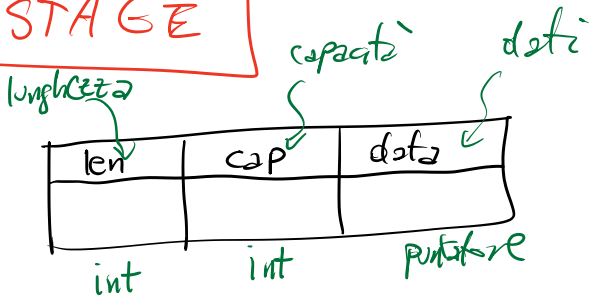
y = x

x = make([]int, 15)



SLICE: IL BACKSTAGE

slice



var x [] int LUNGH. ≤ CAPACITÀ ≥ LUNGH.

x = make([]int, 5, 8)

x[2] = -1

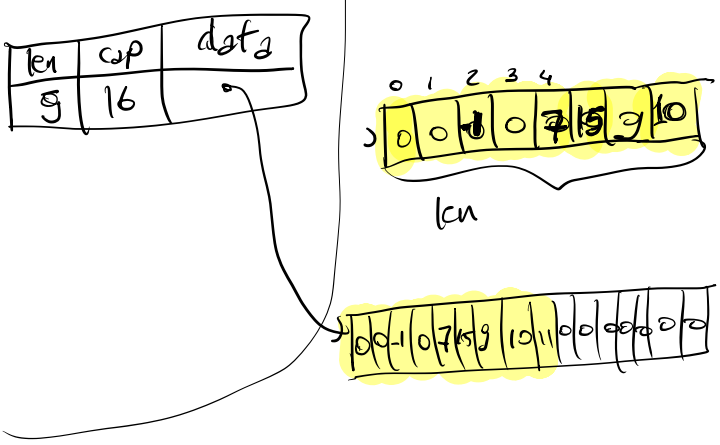
x[4] = 7

x = append(x, 15)

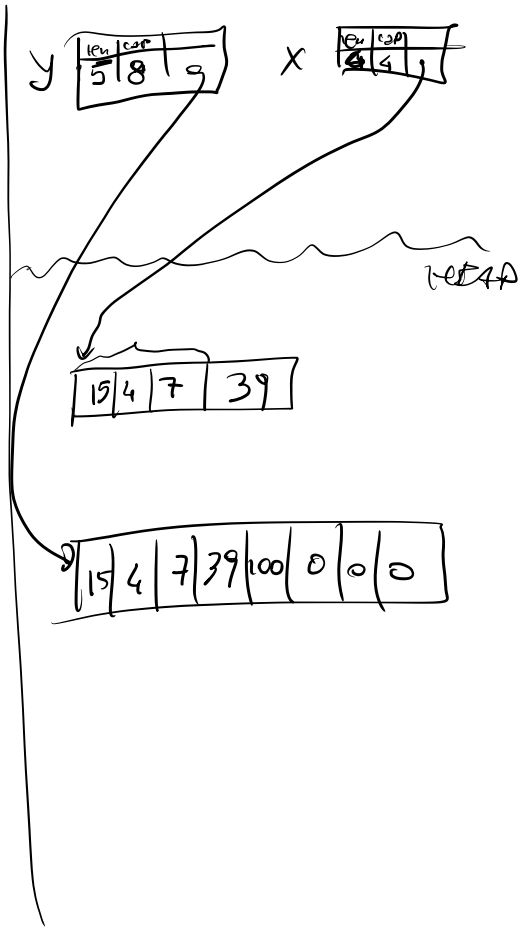
x = append(x, 9)

x = append(x, 10)

x = append(x, 11)

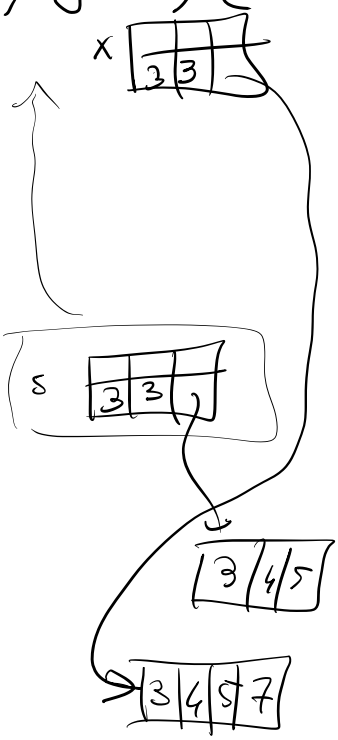


```
var x, y []int
y = make([]int, 3, 4)
for i := 0; i < len(y); i++ {
    y[i] = 3*i+1
}
x = y
y = append(y, 8)
x[0] = 15
x = append(x, 39)
y = append(y, 100)
```



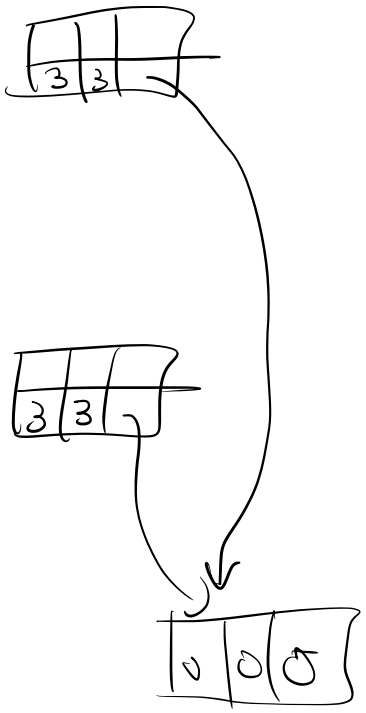
```
func aggiungiUnElemento(x []int, y int) {
    x = append(x, y)
}

func main() {
    s := []int {3,4,5}
    aggiungiUnElemento(s,7)
```



```
func azzeraSlice(x []int) {
    for i:=0; i < len(x); i++ {
        x[i] = 0
    }
}

func main() {
    s := []int{3,4,5}
    azzeraSlice(s)
}
```

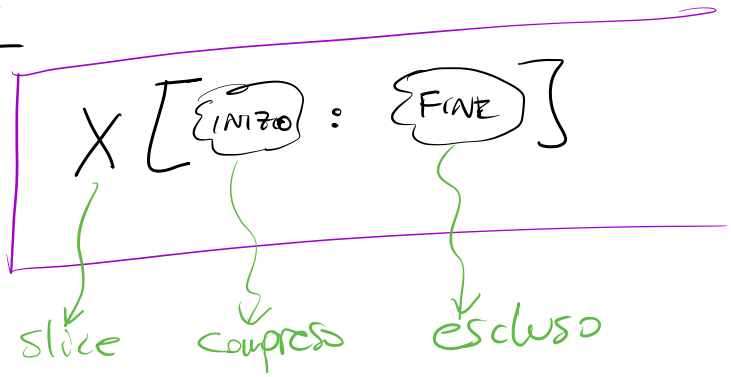


FUNZIONE CHE PRENDE COME ARGOMENTO UNA SLICE

1) LA ACCEDE IN LETTURA ✓
2) MODIFICA DEGLI ELEM. ✓
   MA NON FA APPEND
   ATTENZIONE modifica il par. attuale
3) FA APPEND: DEVE RESTITUIRE LA SLICE OTTENUTA E IL CHIAMANTE LA DEVE RIASSEGNARE ✓ 

SUBSLICING

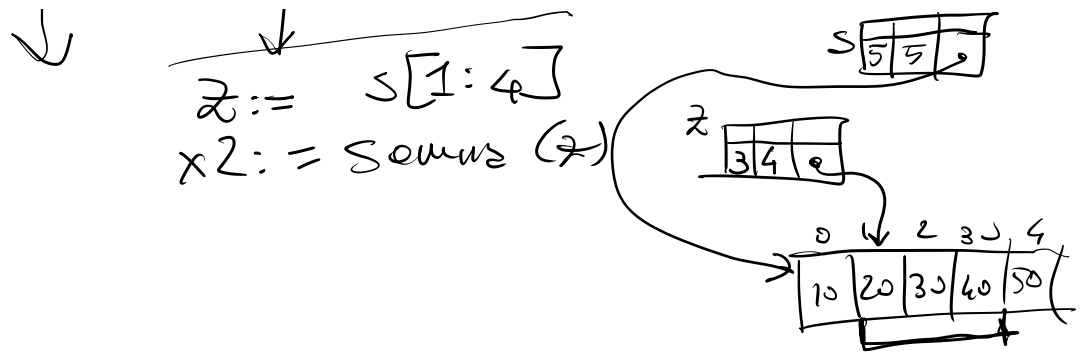
- CREA UNA NUOVA SLICE DA SLICE/~~ARRAY~~ A PARTIRE DA ESISTENTE




```
func somma (x []int) int {
    s:=0
    for _, v := range x {
        s+=v
    }
    return s
}

func main() {
    s := []int {10, 20, 30, 40, 50}
    x1 := somma (s)
    x2 := somma (s[1:4])
```

0 1 2 3 4

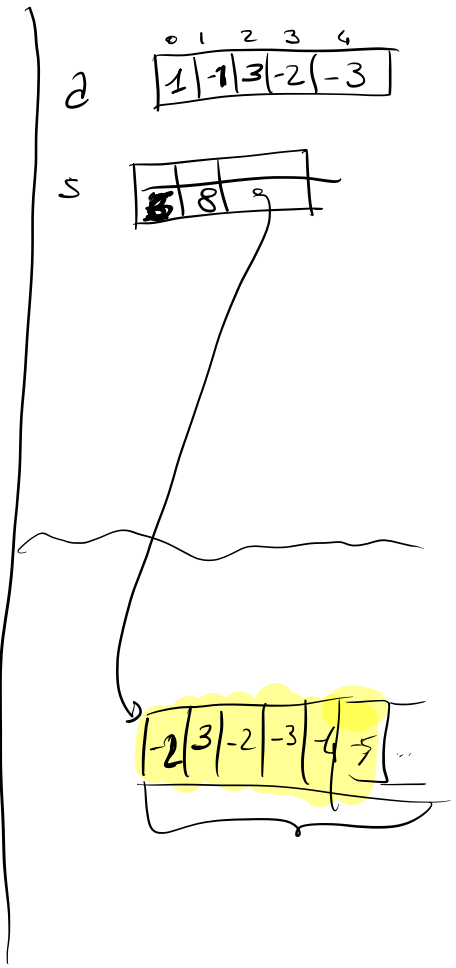


ciao, mamma, domani

```
func join(s []string) string {
    res := ""
    for i := 0; i < len(s)-1; i++ {
        res += s[i] + ","
    }
    res += s[len(s)-1]
    return res
}

s := []string{"ciao", "dozen", "sole"}
fmt.Println(join(s))
fmt.Println(join(s[:len(s)-1]))
fmt.Println(join(s[1:]))
a := [7]string{"LUN", "MAR", "MER",
    "GIO", ---, "DOM"}
fmt.Println(join(a[:5]))
fmt.Println(join(a[:]))
```

```
var a [5]int
a = [5]int {1,2,3,4,5}
var s []int
s = a[1:3]
s[0] = -1
s = append(s, -2)
s = append(s, -3, -4, -5)
s[0] = -2
fmt.Println(a)
fmt.Println(s)
```



- Scrivere una funzione che data due slice di stringhe x e y restituisce il numero di elementi di x che compaiono in y
- Stessa cosa ma con una funzione che non restituisce il numero ma una slice di string
- Scrivere una funzione che data una slice di interi, restituisce una slice di date dei Natali di quegli anni

- Scrive una funzione che data una slice di date restituisce una slice di stringhe che sono i corrispondenti giorni della settimana